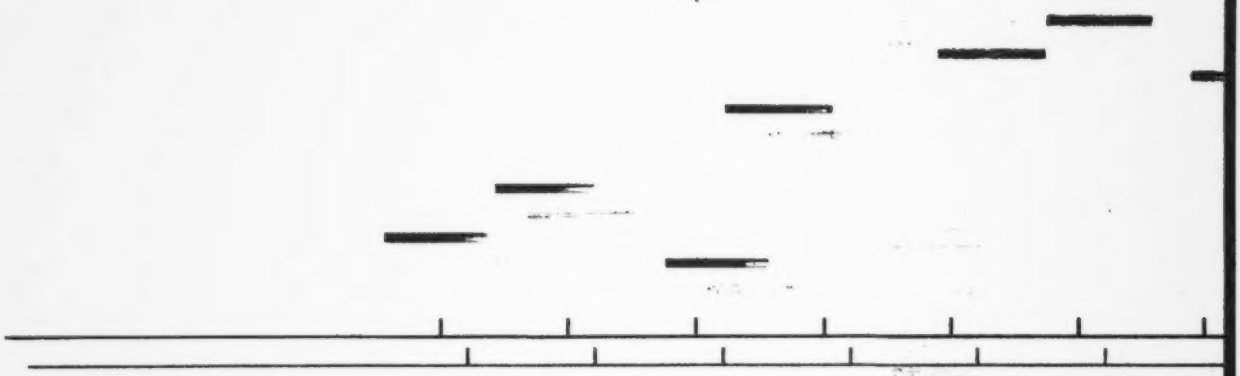
P153385.PDF [Page: 3 of 191]

95-602

C.D.T.
Centre de développement technologique



ÉCOLE
POLYTECHNIQUE

P153385.PDF [Page: 4 of 191]

C.D.T. ÉCOLE POLYTECHNIQUE

– **FINAL REPORT** –

NEURAL NETWORK IMPLEMENTATION STRATEGIES : A COMPARATIVE STUDY

prepared for:

Mr. Pierre Lavoie
DEFENSE SCIENTIST
Electronic Warfare Division
Defense Research Establishment Ottawa
3701 Carling ave., building 75
Ottawa, Ontario, K1A 0Z4

November 1994 C.D.T. Project P1708

P153385.PDF [Page: 5 of 191]

C.D.T.

ÉCOLE POLYTECHNIQUE

Project C.D.T. P1708

– Final Report –

NEURAL NETWORK IMPLEMENTATION STRATEGIES : A COMPARATIVE STUDY

prepared for

DEFENSE SCIENTIST
Electronic Warfare Division
Defense Research Establishment Ottawa
3701 Carling ave., building 75
Ottawa, Ontario, K1A 0Z4

Attention : Mr. Pierre Lavoie

by

Professor Yvon Savaria, eng.
Department of Electrical and Computer Engineering
École Polytechnique de Montréal

submitted by

CENTRE DE DÉVELOPPEMENT TECHNOLOGIQUE (C.D.T.)
ÉCOLE POLYTECHNIQUE DE MONTRÉAL
Campus de l'Université de Montréal
P.O. Box 6079, Station «Centre-ville»
Montréal (Québec) H3C 3A7

November 1994

Yvon Savaria, eng.
Project Director

Neural Network Implementation Strategies: A Comparative Study

Jean-François Crespo and Yvon Savaria

Final progress report

This document reviews the progress made during the sixth, seventh and eight quarters (Febuary 1994 to October 1994). The main tasks of the project were completed on August 5th, 1994, when the final project report was accepted as a masters thesis by a examination jury composed of Professor Jean-Jules Brault, Dr. Pierre Lavoie and Professor Yvon Savaria.

This document is divided in four sections. The first section gives general information on the project. Section two describes the tasks performed during the period under consideration. Section three summarizes the meeting trips and conferences. Finally, section four lists all expenditures for the period previously mentioned.

1.0 General information

1.1 Project schedule

The project was completed on schedule. A final report was accepted as a masters thesis on August 5, 1994 at *École Polytechnique de Montréal* by an examination jury composed of the following members:

- Jean-Jules Brault, Ph.D., president
- Pierre Lavoie, Ph.D., member
- Yvon Savaria, Ph.D., supervisor

This document provides a detailed comparative study of neural network implementation strategies. It focuses on two main classes of implementation technologies, namely the optical based and the VLSI based implementations.

P153385.PDF [Page: 7 of 191]

1.2 Budget

The project is within budget. See section 4.0 on page 3 for more details.

1.3 Assistance or guidance requirements

The scientific authority was regularly consulted and kept informed of the developments and no additional assistance or guidance from the scientific authority or from the government of Canada was required. However, some additional assistance from the scientific authority and Dr. Savaria will be required after October 31st for the completion of a research paper, to be submitted to the *Neural Networks* journal (see section 2.4).

2.0 Progress report

2.1 Task descriptions

This section describes the tasks performed during the nine months beginning Febuary 1 1994 and ending October 31 1994. These tasks are:

- Finalization of research on VLSI implementations of neural networks;
- Writing of final report;
- Preparation for ISCAS 94;
- Writing of a research paper on a modified version of the Fuzzy ART network.

2.2 Finalization of research on VLSI implementations of neural networks

The second part of the research on hardware implementations of neural networks, dealing with electronic implementations was completed. Two types of architectures were considered: digital and analog. Some of the important conclusions were that analog technology offers the best perspective for low power consumption and high levels of integration. On the other hand, digital technology is best suited for precise and flexible architectures, through the use of a digital internal representation of information. Most commercial products available to this date use digital technology. Part of the reason has to do with the still limited maturity of neural network theory, which calls for fast and flexible architectures on which researchers can experiment several different neural network paradigms. Analog implementations seem better suited for dedicated applications.

2.3 Writing of final report

The writing of a final project report was completed near the end of July 1994. It is entitled "Étude sur la réalisation des réseaux de neurones artificiels: comparaison des technologies et architecture du réseau ART1". It is divided in four chapters. Chapter 1 is a review of optical, electronical and neural network theory. Chapter 2 deals with optical implementations of neural networks. Chapter 3 covers electronic implementations. Chapter 4 describes the Direct Coding Rule, a new learning rule developed for ART1 neural networks. For more details, please consult the final report.

2.4 Preparation for ISCAS 94

Preparation for the presentation of a paper entitled "Fast convergence with low precision weights in ART1 networks" was completed. This paper was presented at the 1994 International Symposium on Circuits and Systems (ISCAS) held in London, England, from May 30th to June 2nd.

2.5 Writing of a research paper on a modified version of the Fuzzy ART network

As a follow-up to the paper entitled "Fast Convergence with Low Precision Weights in ART1 Networks", some additional research on Fuzzy ART networks was undertaken. It was identified that Fuzzy ART networks do not respond well to two-way variations of the vigilance parameter. Indeed, when such a network has learned stable templates for a given set of input patterns at a given level of vigilance, coarser clustering is no longer possible when vigilance is lowered.

This limitation was attributed to the internal learning rule of the network: the Weber Law Rule. Indeed, the Weber Law Rule governs the order of search among learned templates. The rule is such that it is biased towards the selection of smaller clusters first, when such clusters exist. When vigilance is lowered, the selection of smaller clusters will prematurely end the searching process, thus hindering its ability to generalize.

Our solution to this problem is the replacement of the Weber Law Rule with the Direct Coding Rule, introduced at ISCAS 94. The learning characteristics of the Direct Coding Rule are such that the size of the clusters first selected depends on the vigilance level. Thus low vigilance means selecting larger clusters first and vice-versa.

Once completed, this paper will be submitted to the *Neural Networks* journal. Expected date of submission is late December 1994.

3.0 Trips and conferences

Mr. Crespo and Dr. Savaria attended the 1994 International Symposium on Circuits and Systems (ISCAS), held in London, England, from May 30th to June 2nd. Mr. Crespo presented an article entitled "Fast Convergence with Low Precision Weights in ART1 Networks".

A meeting between Dr. Lavoie, Dr. Savaria and Mr. Crespo was held on August 4. The purpose of the meeting was to discuss the contents of the research paper on the Modified Learning Law for Fuzzy ART Networks.

4.0 Expenditures

See attached expense statement.

P153385.PDF [Page: 10 of 191]

106-04293/

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE

Ce mémoire intitulé:

ÉTUDE SUR LA RÉALISATION DES RÉSEAUX DE NEURONES ARTIFICIELS: COMPARAISON DES TECHNOLOGIES ET ARCHITECTURE DU RÉSEAU ART1

présenté par Jean-François CRESPO

en vue de l'obtention du grade de Maître ès Sciences Appliquées (M.Sc.A.)

a été dûment accepté par le jury d'examen constitué de:

M. BRAULT, Jean-Jules, Ph.D., président

M. SAVARIA, Yvon, Ph.D., directeur

M. LAVOIE, Pierre, Ph.D., membre

P153385.PDF [Page: 11 of 191]

UNIVERSITÉ DE MONTRÉAL

ÉTUDE SUR LA RÉALISATION DES RÉSEAUX DE NEURONES ARTIFICIELS: COMPARAISON DES TECHNOLOGIES ET ARCHITECTURE DU RÉSEAU ART1

par

JEAN-FRANÇOIS CRESPO
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE ET DE GÉNIE INFORMATIQUE
ÉCOLE POLYTECHNIQUE

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU GRADE DE MAÎTRE ÈS SCIENCES APPLIQUÉES (M.Sc.A.)
(GÉNIE ÉLECTRIQUE)
Août 1994

© droits réservés de Jean-François CRESPO 1994

SOMMAIRE

Ce mémoire comporte deux volets. Le premier est une étude comparative des technologies pour la réalisation des réseaux de neurones artificiels. Les technologies considérées sont l'optique et l'électronique. Pour l'optique, nous considérons les architectures à processeurs algébriques optiques, à corrélateurs optiques et à interconnexions holographiques. Pour l'électronique, nous considérons les architectures numériques et analogiques. Les deux technologies sont confrontées selon trois types de critères; la performance, les contraintes technologiques et les considérations d'ordre pratique. L'optique permet d'atteindre les performances les plus élevées avec les architectures à corrélateurs optiques, ces dernières tirant profit du pouvoir d'interconnexion supérieur de la technologie. Cependant la faible maturité de la technologie ralentit considérablement sa commercialisation. L'électronique numérique surclasse les autres architectures en matière de flexibilité et d'exactitude dans les calculs. Elle possède également la plus grande part du marché des neuro-ordinateurs. L'électronique analogique en mode infraseuil offre les meilleures perspectives en terme d'économie de puissance.

Dans le cadre du deuxième volet, une nouvelle loi d'apprentissage, la *Loi de Codage Direct*, est proposée pour l'apprentissage dans les réseaux *Adaptive Resonance Theory* (ART). Cette loi nécessite moins de précision de calcul que la *Loi de Weber* traditionnelle et elle modifie la dynamique de recherche dans le but d'accélérer la convergence. Après une brève analyse mathématique de la nouvelle loi, un réseau ART1 faisant appel à cette loi est appliqué à un problème de détection passive de signaux radar. Des résultats de simulations démontrent qu'à un niveau de quantification des poids de 7 bits, la Loi de Codage Direct accélère la convergence du réseau par un facteur 10 par rapport à la Loi de Weber.

ABSTRACT

This work is divided into two parts. Part one is a comparative study of the implementation technologies for artificial neural networks. The technologies considered are optics and electronics. For the optics technology, we consider optical algebraic processors based architectures, correlator based architectures and holographic interconnection based architectures. For the electronics technology, we consider digital and analog architectures. Both technologies are confronted on the basis of performance, technological and practicality criterias. Optics allows for the highest performances for correlator based architectures which take full advantage of the superior interconnecting power of that technology. However, the limited maturity of optics hinders commercialization. Digital electronics outperforms the other architectures because of its flexibility and the accuracy of computation it provides. It is also the most successfully commercialized approach. However, subthreshold analog electronics is the most power-efficient approach.

In part two, a new learning law, the *Direct Coding Rule*, is proposed for bottom-up long term memory learning in Adaptive Resonance Theory (ART) networks. This law requires less computational precision than the traditional *Weber Law Rule* and modifies the search dynamics of the network to accelerate convergence. Following a brief mathematical analysis of the new learning law, an ART1 network based on this law is applied to a passive radar detection problem. Simulation results show that for a quantization level of 7 bits in the weights, the Direct Coding Rule speeds up convergence by a factor of 10 compared to the Weber Law Rule.

REMERCIEMENTS

Je tiens tout d'abord à remercier mon directeur de recherche, Dr. Yvon Savaria, pour m'avoir supervisé durant ces deux dernières années. Malgré son statut de directeur et sa solide réputation dans le domaine de l'ITGE, Yvon n'éprouve aucune difficulté à considérer ses étudiants comme des *collègues*, ce qui, en soi, est un moteur d'inspirations. Mes remerciements vont ensuite au Dr. Pierre Lavoie du CRDO, pour avoir cru en mes capacités de mener ce projet à terme et pour m'avoir initié à l'univers des réseaux de neurones artificiels.

Je tiens ensuite à remercier l'Institut National d'Optique, en particulier le Dr. Denis Gingras, pour m'avoir permis de suivre un stage à l'INO et pour m'avoir guidé dans ma recherche sur les réalisations optiques de réseaux de neurones. Des remerciements vont ensuite au reste du personnel de l'INO, en particulier à (en ordre alphabétique) Alain Bergeron, Michel Bouchard, Michel Doucet, Eric Eustache et Jonny Gauvin.

Je désire également exprimer ma gratitude envers le Centre de Recherches pour la Défense Ottawa (CRDO) d'une part pour m'avoir fourni les données radar nécessaires pour les simulations du chapitre V, et d'autre part pour le support matériel et financier. Je remercie également le Conseil de Recherches en Sciences Naturelles et Génie (CRSNG) pour l'octroi de ma bourse d'études supérieures.

Finalement, il serait injuste de passer sous silence le support moral de mes parents et amis, en particulier celui de ma conjointe, Dominique. Je remercie également Dominique pour son aide précieuse lors de la révision finale de ce mémoire.

P153385.PDF [Page: 15 of 191]

TABLE DES MATIÈRES

SOMMAIRE iii

ABSTRACT iv

REMERCIEMENTS v

TABLE DES MATIÈRES vi

LISTE DES FIGURES ix

LISTE DES TABLEAUX xii

AVANT-PROPOS xiii

INTRODUCTION 1

CHAPITRE I ÉLÉMENTS DE THÉORIE 5

1.1 Éléments de théorie en réseaux de neurones artificiels 5
1.1.1 Modèle classique du neurone 6
1.1.2 Réseaux de neurones 6
1.1.3 Apprentissage dans les réseaux de neurones 8
1.2 Éléments de théorie en optique 9
1.2.1 Nature ondulatoire de la lumière 9
1.2.2 Cohérence spatiale 11
1.2.3 Phénomène de diffraction 12
1.2.4 Holographie 13
1.3 Éléments de théorie en électronique 15
1.3.1 Le transistor à effet de champ MOSFET 15
1.3.2 Modes d'utilisation du transistor MOSFET 18
1.3.3 Les circuits de base 20

P153385.PDF [Page: 16 of 191]

CHAPITRE II RÉSEAUX DE NEURONES OPTIQUES 25

2.1 Composants optiques 26
2.1.1 Lentilles 26
2.1.2 Émetteurs 28
2.1.3 Détecteurs 29
2.1.4 Modulateurs spatiaux de lumière 30
2.1.5 Hologrammes 31
2.2 Architectures à processeurs algébriques optiques 32
2.2.1 Principe 32
2.2.2 Le multiplieur vecteur-matrice optique 32
2.2.3 Considérations d'ordre pratique 34
2.2.4 Performances 38
2.3 Architectures à corrélateurs optiques 42
2.3.1 Le filtre de Vander Lugt 43
2.3.2 Considérations d'ordre pratique 46
2.3.3 Performances 48
2.4 Architectures à interconnexions holographiques 54
2.4.1 Principe 54
2.4.2 Topologies d'interconnexion et degrés de liberté 55
2.4.3 Multiplexage spatial 57
2.4.4 Multiplexage volumique 58
2.4.5 Considérations d'ordre pratique 59
2.4.6 Performances 62

CHAPITRE III RÉSEAUX DE NEURONES ÉLECTRONIQUES 66

3.1 Complexité dans les réseaux de neurones électroniques 66
3.2 Architectures numériques 71
3.2.1 Principe 72
3.2.2 Considérations d'ordre pratique 76
3.2.3 Performances 77
3.3 Architectures analogiques 83
3.3.1 Principe 83
3.3.2 Considérations d'ordre pratique 85
3.3.2.1 Mémoires numériques contrôlant une valeur analogique quantifiée 86
3.3.2.2 Condensateurs 87
3.3.2.3 Mémoires non-volatiles 89
3.3.2.4 Mémoires analogiques actives 92
3.3.3 Performances 93

P153385.PDF [Page: 17 of 191]

CHAPITRE IV **COMPARAISON OPTIQUE/ÉLECTRONIQUE** **99**

4.1 Critères de performance 100
4.1.1 Vitesse 100
4.1.2 Capacité d'interconnexion 102
4.1.3 Précision et exactitude dans les calculs 103
4.2 Critères technologiques 107
4.2.1 Fiabilité et tolérance aux pannes 107
4.2.2 Intégration 108
4.2.3 Mise à l'échelle 110
4.2.4 Flexibilité 112
4.3 Critères d'ordre pratique 115
4.3.1 Consommation de puissance 115
4.3.2 Poids et dimensions physiques 118
4.3.3 Coût économique et maturité 119

CHAPITRE V **ÉTUDE DE CAS: RÉALISATION DU RÉSEAU ART1** **121**

5.1 Introduction 122
5.2 Principe d'opération du réseau ART1 et la Loi de Weber 123
5.3 Loi de Codage Direct 127
5.4 Résultats de simulations 132

CONCLUSION 139

BIBLIOGRAPHIE 144

ANNEXE A 153

LISTE DES FIGURES

Figure 1.1: Schéma de a) un neurone typique et b) sa fonction de seuillage non-linéaire. 7

Figure 1.2: Réseau de neurones à deux couches sans rétroaction. 7

Figure 1.3: Phénomène de diffraction pour une grille diffractive. a) Grille diffractive; b) Modélisation par une colonne de sources radiales; c) Condition d'interférence constructive. 13

Figure 1.4: Reconstruction holographique. 15

Figure 1.5: Miroirs de courant a) de type *n*; b) de type *p*. 20

Figure 1.6: Amplificateur différentiel de transconductance. 21

Figure 1.7: Multiplieur analogique à quatre quadrants. 23

Figure 1.8: Multiplieur quatre quadrants de Gilbert 23

Figure 2.1: Exemple de lentilles. a) Sphérique; b) Cylindrique. 26

Figure 2.2: Taxinomie des modulateurs spatiaux de lumière utilisés en traitement optique de l'information. 30

Figure 2.3: Multiplieur vecteur-matrice optique. 33

Figure 2.4: Réseaux de neurones à noyau algébrique optique. a) Avec rétroaction (ex. Hopfield); b) Perceptron multicouche (ex. backpropagation). 34

Figure 2.5: Exemple d'aberration sphérique; a) image d'entrée; b) image de sortie. 37

Figure 2.6: Performances pour architectures à processeur algébrique optique utilisant des TVCL. 40

Figure 2.7: Performances pour architectures à processeurs algébriques optiques utilisant des MSL acousto-optiques. 42

P153385.PDF [Page: 19 of 191]

Figure 2.8: Montage d'un système de filtrage optique. 44

Figure 2.9: Principe d'opération de la mémoire associative de Psaltis (d'après [44]). 50

Figure 2.10: Métriques de performance pour architectures à corrélateurs optiques utilisant des valves optiques. 52

Figure 2.11: Principe d'interconnexion holographique. 54

Figure 2.12: Montage optique du système d'interconnexion de Caulfield 58

Figure 2.13: Montage optique du système d'interconnexion de Psaltis [68]. 60

Figure 3.1: Topologie d'interconnexion de complexité $O(N^3)$ 68

Figure 3.2: Topologie numérique de complexité $O(N^2(\log_2 N)^2)$; a) Schéma d'interconnexion; b) Arbre de sommateurs replié: la complexité est déterminée par les fils. 69

Figure 3.3: Schéma de l'architecture CNAPS de Hammerstrom. Le cadre gris représente les frontières d'une puce 74

Figure 3.4: Schéma d'une architecture systolique circulaire (d'après [51]). 75

Figure 3.5: Graphique des performances des architectures numériques décrites dans le texte. 82

Figure 3.6: Schéma type d'une architecture analogique. 84

Figure 3.7: Génération d'un courant analogique à partir d'une valeur numérique. 86

Figure 3.8: Stockage d'une tension analogique à l'aide d'un condensateur (adapté de [42]). 87

Figure 3.9: Synapse analogique à mémoire différentielle (d'après [34]). 89

Figure 3.10: Vue en coupe a) d'un transistor MNOS à canal n (d'après [42]); b) d'un transistor FLOTOX d'Intel (d'après [42]). 90

Figure 3.11: Principe de fonctionnement d'une boucle A/N-N/A (d'après [42]). 93

Figure 3.12: Technique de rafraîchissement en escalier (d'après [79]) 94

Figure 5.1: Schéma du réseau ART1. 124

P153385.PDF [Page: 20 of 191]

Figure 5.2: Famille de courbes d'ordonnancement pour $N = 16$ et $\Psi = 32$.....130

Figure 5.3: Code a) thermomètre et b) rampant. 132

Figure 5.4: Temps de convergence, mesurés en itérations de recherche, pour les cinq réseaux ART1.....134

Figure 5.5: Scores obtenus pour les mises en grappes des cinq réseaux ART1.....138

P153385.PDF [Page: 21 of 191]

LISTE DES TABLEAUX

Tableau 2.1: Caractéristiques de quelques téléviseurs à cristaux liquides ([21]). 39

Tableau 2.2: Performances de quelques MVM acousto-optiques utilisés dans des applications de réseaux de neurones (adapté de [29]). 41

Tableau 2.3: Caractéristiques de quelques valves optiques pour la mise en œuvre d'architectures à corrélateurs optiques. 52

Tableau 2.4: Résumé des avantages et inconvénients des trois types d'architectures optiques selon les points de vue pratique et performance. 65

Tableau 3.1: Vitesses de diverses architectures de RNA numériques. 82

Tableau 3.2: Résumé des performances des architectures analogiques décrites dans le texte. 97

Tableau 3.3: Résumé des avantages et inconvénients des architectures électroniques numériques et analogiques selon les points de vue pratique et performance. 98

Tableau 4.1: Résumé des vitesses des architectures optiques et électroniques. Toutes les données sont en connexions/seconde. 101

Tableau 5.1: Tableau de contingence. 136

AVANT-PROPOS

Ce projet a été rendu possible grâce à un financement du Ministère de la Défense du Canada, Centre de Recherche pour la Défense Ottawa (CRDO) dans le cadre du contrat no. W7714-2-9630/01-ST et d'une collaboration spéciale avec l'Institut National d'Optique, Sainte-Foy. L'auteur était financé en partie par le Conseil National de Recherches en Sciences Naturelles et en Génie (CRSNG).

INTRODUCTION

Nous pouvons retracer les débuts de la recherche sur les réseaux de neurones artificiels (RNA) jusqu'en 1943, année de publication de l'article de Warren McCulloch et Walter Pitts. Ce premier article proposait alors un modèle mathématique primitif du traitement neuronal. Par la suite, Frank Rosenblatt proposa en 1957 un nouveau modèle plus poussé qu'il baptisa le *perceptron*. Avec l'aide de son collègue Charles Wightman, il bâtit le premier neuro-ordinateur de l'histoire: le *Mark I Perceptron*, qu'il utilisa avec succès pour une application de reconnaissance de caractères typographiques. Ces développements ont vite fait de provoquer un profond émoi auprès de la population qui anticipait déjà le jour imminent où des robots intelligents, capables d'accepter des ordres verbaux, seraient au service de l'humanité. Malheureusement, cet enthousiasme s'estompa rapidement lorsque Marven Minsky et Seymour Papert entreprirent, au milieu des années 60, une campagne de dissuasion dans le but de discréditer la recherche en RNA. Leur principal argument était que le perceptron était incapable de réaliser des primitives de calcul essentielles comme la fonction XOR. Il s'est avéré plus tard que la solution à ce problème consistait tout simplement à utiliser deux couches de perceptrons. Néanmoins, leur campagne de dissuasion fut un succès et eu pour effet de plonger le domaine de recherche des RNA dans une profonde léthargie de 1967 à 1982. Ce n'est qu'en 1983 qu'Ira Skurnick, alors un des dirigeants du *Defence Advanced Research Projects Agency* (DARPA), alla à l'encon-

P153385.PDF [Page: 24 of 191]

tre des tendances populaires et accepta de financer plusieurs projets de RNA. Entre 1983 et 1986, le physicien John Hopfield, dont la réputation était déjà solide, s'intéressa également aux RNA et publia deux articles clés qui, combinés au courage de Skurnick, contribuèrent à mener graduellement ce domaine de recherche à son niveau de notoriété actuel.

Il est sans doute très probable que l'essor de la recherche en RNA fut accentué par les récents progrès en intégration à très grande échelle (ITGE). Jusqu'à lors, les chercheurs qui désiraient émuler une architecture neuronale devaient se contenter d'écrire un programme séquentiel pour un ordinateur de type Von-Neumann. Ces chercheurs se sont vite rendus compte que s'ils voulaient éviter que le domaine de recherche ne retombe à nouveau dans une léthargie, ils devraient se pencher sérieusement sur le problème de la réalisation matérielle des réseaux de neurones artificiels. Ce besoin a donné naissance à deux écoles de pensées, l'une prônant la mise en œuvre électronique de RNA, l'autre prônant la mise en œuvre optique. Depuis les dix dernières années, une foule d'architectures optiques et électroniques ont été proposées par plusieurs chercheurs à travers le monde, certaines aboutissant même à la commercialisation. Il existe, dans la vaste littérature sur les RNA, plusieurs synthèses [12][34][38][54], mais à notre connaissance, il n'existe aucune étude comparative entre l'optique et l'électronique se basant de façon objective et systématique sur plusieurs critères discriminants. Le but de cet ouvrage est donc de comparer de façon objective ces deux approches. Nous procéderons également à l'étude du cas de la réalisation matérielle du réseau *Adaptive Resonance Theory 1*(ART1) en technologie électronique.

Organisation du mémoire

Cet ouvrage comporte cinq chapitres. Dans le premier chapitre, nous révisons d'abord les notions de base en réseaux de neurones artificiels, en optique et en électronique, dans

P153385.PDF [Page: 25 of 191]

le but de faciliter la compréhension des chapitres subséquents. Le principe d'opération du neurone classique, les diverses topologies d'interconnexion ainsi que le concept d'apprentissage seront présentés. La branche de l'optique qui nous est pertinente est l'*optique diffractive*. Par conséquent, après un rappel sur la nature ondulatoire de la lumière, la notion de cohérence spatiale sera abordée avant de décrire le phénomène fondamental de la diffraction. Ces rappels servent de prémisse pour introduire le principe de l'holographie, sans doute le concept le plus important en optique diffractive. Par la suite, les notions de base en électronique sont abordées. Le constituant majeur de la plupart des circuits électroniques, le transistor MOSFET, est d'abord décrit et quelques-uns de ses principaux modes d'utilisation sont énumérés. Finalement, plusieurs circuits de base sont expliqués. Ces circuits forment le coeur des architectures électroniques de RNA.

Le chapitre deux entame la description des réalisations matérielles par les réalisations optiques. Étant donné qu'il est fort probable que le lecteur soit davantage familier avec l'électronique qu'avec l'optique, la section 2.1 décrit les principaux composants utilisés en optique, soit les lentilles, les émetteurs, les détecteurs, les modulateurs spatiaux de lumière et les hologrammes. Trois grandes classes d'architectures optiques sont ensuite proposées et décrites: les architectures à processeurs algébriques optiques, les architectures à corrélateurs optiques et les architectures à interconnexions holographiques. Pour chacune de ces architectures, le principe est d'abord décrit, puis nous discutons par la suite de certaines considérations d'ordre pratique ayant trait à leur mise en œuvre. Finalement, leurs performances sont calculées, en mettant l'accent sur la vitesse.

Le chapitre trois poursuit la description des réalisations matérielles en décrivant les réalisations électroniques des RNA. Ces architectures se divisent en deux grandes classes: les architectures numériques et analogiques. Mais avant de décrire ces deux classes, la section 3.1 traite d'abord de la complexité dans les réseaux de neurones électroniques.

P153385.PDF [Page: 26 of 191]

Cette section a pour but de justifier les choix de conception selon l'approche choisie. La structure du reste du chapitre est identique à celle du chapitre deux: pour chaque approche, le principe est d'abord décrit, puis les considérations d'ordre pratique sont investiguées pour ensuite terminer avec la compilation des performances.

Les chapitres deux et trois étant de nature plutôt descriptive, avec quelques conclusions d'ordre pratique et une emphase sur la vitesse des architectures, le chapitre quatre examine d'autres critères pour permettre une comparaison adéquate des deux technologies. Ces critères sont regroupés en trois classes: les critères de performance, les critères technologiques et les critères d'ordre pratique. Les critères d'ordre pratique sont la vitesse, la capacité d'interconnexion, la précision et l'exactitude des calculs. Parmi les critères technologiques, on retrouve la fiabilité et la tolérance aux pannes, l'intégration, la mise à l'échelle et la flexibilité. Finalement, les critères d'ordre pratique sont la consommation de puissance, le poids et les dimensions physiques, le coût économique et la maturité.

À titre d'exemple, le chapitre cinq est une étude de cas de la réalisation d'un réseau de neurone spécifique à l'aide de la technologie électronique. Ce réseau a pour nom *Adaptive Resonance Theory 1* (ART1). Le chapitre traite uniquement d'une modification apportée à l'architecture ART1 dans le but ultime de permettre une réalisation matérielle plus économique et plus performante. Cette modification porte le nom de *Loi de Codage Direct*. Des résultats de simulation pour une application en mesures de soutien électronique sont présentés en fin de chapitre. Un court rapport sur une proposition de mise en œuvre est incluse en annexe.

CHAPITRE I ÉLÉMENTS DE THÉORIE

Dans ce premier chapitre, les notions de base en réseaux de neurones artificiels, en optique et en électronique sont révisées. Cette révision sommaire a pour but de faciliter la compréhension des nombreuses architectures qui seront décrites dans les chapitres II et III. La première section est un rappel des notions de base en réseaux de neurones artificiels. Le principe d'opération du neurone classique, les diverses topologies d'interconnexion ainsi que le concept d'apprentissage seront présentés. La deuxième section traite des notions de base en optique. Après un bref rappel sur la nature ondulatoire de la lumière, les concepts de cohérence spatiale, de diffraction, ainsi que d'holographie sont décrits. La troisième section porte sur les notions de base en électronique. Les comportements du transistor MOSFET en mode supraseuil et infraseuil sont d'abord introduits. Par après, les principaux modes d'utilisation du transistor sont présentés. Finalement, quelques circuits analogiques de base sont décrits. Notez que les circuits numériques de base ne seront pas couverts, leur compréhension étant en général plus solide que celle des circuits analogiques.

1.1 Éléments de théorie en réseaux de neurones artificiels

Dans cette section, nous révisons les notions de base en réseaux de neurones artificiels (RNA). Le domaine de recherche des RNA est très vaste et un certain nombre de volumes

P153385.PDF [Page: 28 of 191]

d'introduction ont déjà été publiés sur le sujet [38][54]. Par conséquent, notre révision sera très brève et elle aura pour seul but de fixer la notation et la terminologie pour faciliter la lecture des chapitres subséquents.

1.1.1 Modèle classique du neurone

Le modèle universellement accepté du neurone artificiel est celui d'un processeur élémentaire qui effectue une somme pondérée de ses signaux d'entrée. Le signal de sortie du processeur est ensuite produit en appliquant une fonction de seuillage non-linéaire (bien souvent de forme sigmoïdale). Supposons qu'un neurone j donné possède N entrants, dénotés par $x_i, i = 1, 2, ..., N$. Dénotons par $w_{ij}, i = 1, 2, ..., N$ les N facteurs de pondération des entrants x_i (le facteur w_{ij} pondère l'entrant d'indice x_i). La somme pondérée t_j, connue sous le nom d'*entrant net*, est donc définie par l'équation suivante:

$$t_j = \sum_{i=1}^{N} x_i w_{ij} \quad . \qquad (1.1)$$

Une fois l'entrant net calculé, le neurone applique cette valeur à une fonction de seuillage non-linéaire $f(\alpha)$. La sortie, dénotée par s_j, est la *valeur d'activation* (ou *activation*) du neurone:

$$s_j = f\{t_j\} = f\{\sum_{i=1}^{N} x_i w_{ij}\} . \qquad (1.2)$$

La figure 1.1 illustre schématiquement un neurone et un exemple de fonction de seuillage.

1.1.2 Réseaux de neurones

Plusieurs neurones interconnectés forment ce qu'il convient d'appeler un *réseau de neurones*. Selon la topologie d'interconnexion, on distingue deux classes principales de réseaux: les réseaux avec rétroaction (*feedback*) et sans rétroaction (*feedforward*). La

P153385.PDF [Page: 29 of 191]

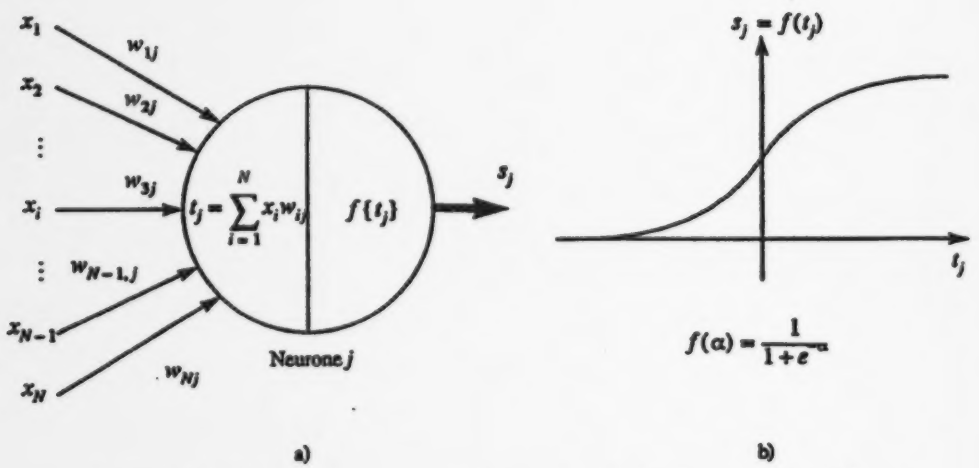


Figure 1.1: Schéma de a) un neurone typique et b) sa fonction de seuillage non-linéaire.

figure 1.2 illustre un exemple d'un réseau sans rétroaction. Ce dernier est constitué de deux *couches* de neurones, la couche du bas (appelée *couche d'entrée*) étant complètement interconnectée avec la couche du haut (appelée *couche de sortie*). La première couche comporte N neurones et la deuxième couche, M neurones. Notez que le rôle de la couche d'entrée est uniquement de propager chaque signal d'entrée à tous les neurones de la couche de sortie.

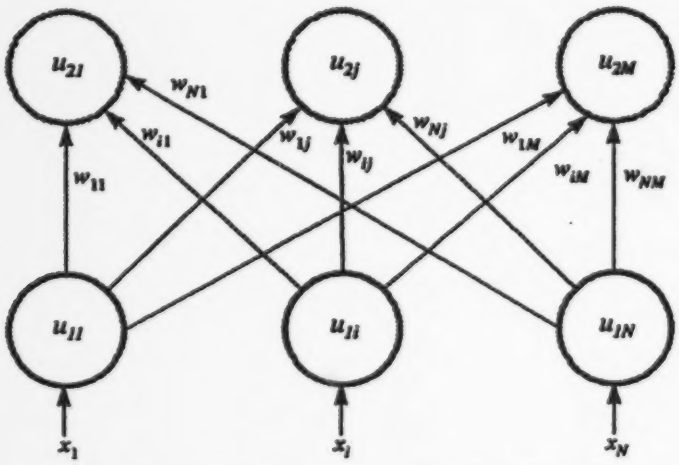


Figure 1.2: Réseau de neurones à deux couches sans rétroaction.

P153385.PDF [Page: 30 of 191]

Dans un réseau de neurones, les facteurs de pondération sont souvent appelés *poids d'interconnexion* puisqu'ils pondèrent les signaux se propageant dans les interconnexions. Ainsi, w_{ij} est le poids d'interconnexion entre le neurone i de la couche d'entrée et le neurone j de la couche de sortie.

Il est souvent utile d'adopter la notation matricielle pour décrire l'opération d'un réseau de neurones. Si nous dénotons le vecteur d'entrée par $\mathbf{X} = (x_1, x_2, ..., x_N)$, le vecteur d'entrants nets par $\mathbf{T} = (t_1, t_2, ..., t_M)$, et la matrice des poids par

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1M} \\ w_{21} & w_{22} & \cdots & w_{2M} \\ \cdots & \cdots & \cdots & \cdots \\ w_{N1} & w_{N2} & \cdots & w_{NM} \end{bmatrix}, \tag{1.3}$$

alors l'opération du réseau peut être décrite avec les deux équations matricielles suivantes:

$$\mathbf{T} = \mathbf{XW}, \tag{1.4}$$

$$\mathbf{S} = f(\mathbf{T}), \tag{1.5}$$

où nous assumons que la fonction de seuillage $f(\alpha)$ est appliquée à chaque élément du vecteur T.

1.1.3 Apprentissage dans les réseaux de neurones

Une caractéristique importante des réseaux de neurones est leur capacité d'apprendre. Cet apprentissage s'opère par une modification des poids d'interconnexion. La loi qui régit la modification des poids s'appelle la *loi d'apprentissage*. Il existe plusieurs lois d'apprentissage, chacune ayant pour but de permettre au réseau d'assumer un certain comportement désiré. Par exemple, la loi *backpropagation* permet d'associer un certain vecteur Y au vecteur d'entrée X en ajustant les poids d'interconnexion d'un réseau comme celui de la figure 1.2 jusqu'à ce que l'erreur quadratique entre Y et la sortie du réseau S

P153385.PDF [Page: 31 of 191]

soit en dessous d'un certain seuil. Pour plus de détails sur les différentes lois d'apprentissage, le lecteur peut consulter les références [38] et [54].

1.2 Éléments de théorie en optique

Cette section est une révision des principes et notions de base en physique optique. Nous expliquons d'abord brièvement le modèle ondulatoire de la lumière. Par la suite, la notion de cohérence spatiale est clarifiée pour ensuite aborder la diffraction, puis finalement l'holographie.

1.2.1 Nature ondulatoire de la lumière

Il existe deux modèles pour décrire le comportement de la lumière. Le premier modèle, appelé *modèle particulaire*, décrit la lumière comme étant le déplacement de particules élémentaires appelées *photons*. Cependant, ce modèle, à lui seul, ne peut rendre compte de certains phénomènes comme la diffraction. En effet, si un faisceau lumineux éclaire avec incidence perpendiculaire un plan comportant une très petite ouverture circulaire, le faisceau à la sortie de l'ouverture divergera légèrement. Ce phénomène ne peut clairement pas être expliqué si l'on ne considère que le modèle particulaire. C'est pourquoi il existe un second modèle: le modèle *ondulatoire*. En optique diffractive, nous faisons appel presque exclusivement à ce modèle. Le reste de cette sous-section présente les grandes lignes de ce modèle.

Toute onde, en particulier une onde lumineuse, peut être décrite mathématiquement par l'équation suivante:

$$y(x,t) = A\cos(\omega t - kx + \varphi)\ , \tag{1.6}$$

où A est l'amplitude de l'onde, ω, la fréquence angulaire, exprimée en radians/seconde, t le temps, k, un constante fonction de la longueur d'onde de la lumière, x la position et φ,

P153385.PDF [Page: 32 of 191]

un déphasage initial quelconque. La fréquence angulaire ω est reliée à la fréquence temporelle f par la relation

$$\omega = 2\pi f = 2\pi/T, \tag{1.7}$$

où T est la période de l'onde, soit le temps (en secondes) entre deux maximums pour une position fixe.

De la même façon, le nombre k est relié à la fréquence spatiale υ par la relation

$$k = 2\pi\upsilon = 2\pi/\lambda, \tag{1.8}$$

où λ est la longueur d'onde, soit la distance entre deux maximums pour un temps fixe.

La longueur d'onde et la fréquence temporelle sont reliées par la relation suivante:

$$\lambda = \frac{c}{f}, \tag{1.9}$$

où c est la vitesse de la lumière. Un faisceau de lumière blanche, i.e. à large bande spectrale, peut être décomposé en une série de «couleurs», chacune correspondant à une longueur d'onde spécifique.

Il est souvent utile d'adopter la notation de phaseur pour représenter une onde lumineuse. Ainsi, par la relation d'Euler,

$$y(x,t) = A\cos(\omega t - kx + \varphi) = \mathrm{Re}\{Ae^{j(\omega t - kx + \varphi)}\}, \tag{1.10}$$

où *Re* représente la partie réelle d'un nombre complexe.

Le terme ωt est souvent omis, car bien souvent nous avons affaire à des ondes dites *stationnaires* pour lesquelles le temps n'influence pas la position des nœuds. De plus, en optique, nous ne considérons que les signaux analytiques, i.e. ceux dont le spectre est nul

P153385.PDF [Page: 33 of 191]

pour des fréquences spatiales négatives. Par conséquent, le signal analytique d'un cosinus (équation (1.10)) est une exponentielle complexe de la forme

$$y(x) = Ae^{j(-kx+\varphi)} . \tag{1.11}$$

1.2.2 Cohérence spatiale

Lorsque plusieurs ondes lumineuses oscillent en phase, leurs contributions s'additionnent pour donner naissance à un faisceau sinusoïdal *cohérent spatialement*. La *longueur de cohérence* est définie comme la distance de parcours de la lumière le long de laquelle la phase relative entre deux faisceaux est constante. La cohérence spatiale revêt une importance capitale en optique diffractive, car elle est à la base du phénomène de diffraction.

Lorsque deux faisceaux individuellement cohérents se rencontrent sur une pellicule photosensible, leur interaction résulte en un patron d'interférence qui peut être enregistré par cette pellicule. Pour illustrer ce phénomène, supposons qu'une onde $y_1(x) = e^{jk_1x}$ rencontre une deuxième onde $y_2(x) = e^{jk_2x}$ (nous avons posé comme hypothèse simplificatrice que les déphasages φ pour les deux ondes sont nuls). Comme les ondes sont toutes deux individuellement cohérentes, l'onde résultante sera leur somme complexe. Ainsi,

$$z(x) = y_1(x) + y_2(x) = e^{jk_1x} + e^{jk_2x} . \tag{1.12}$$

Une pellicule photosensible ne répond qu'à l'intensité lumineuse, elle-même proportionnelle au carré de l'amplitude de l'onde résultante, soit

$$I(x) = \left|e^{jk_1x} + e^{jk_2x}\right|^2 \tag{1.13}$$

$$= \left|e^{jk_1x}\right|^2 + \left|e^{jk_2x}\right|^2 + e^{jk_1x}\left(e^{jk_2x}\right)^* + \left(e^{jk_1x}\right)^* e^{jk_2x} \tag{1.14}$$

$$= 1 + 1 + e^{j(k_1-k_2)x} + e^{-j(k_1-k_2)x} \tag{1.15}$$

$$I(x) = 2 + 2\cos\left[(k_1 - k_2)x\right], \tag{1.16}$$

où le conjugué complexe est dénoté par le symbole * en exposant.

P153385.PDF [Page: 34 of 191]

La distribution de l'intensité lumineuse sur la pellicule photosensible est donc sinusoïdale de fréquence spatiale $k_1 - k_2$. La patron sur le film, une fois développé, ressemblera à la figure 1.3a de la page suivante, si on remplace x par y dans l'équation (1.16). Notez que cette intensité est toujours positive (une intensité négative n'a pas de sens physique), d'où l'apparition du biais d'intensité de 2 pour compenser l'apport négatif du cosinus.

1.2.3 Phénomène de diffraction

Avant d'aborder le sujet de l'holographie, nous allons tout d'abord expliquer le phénomène de diffraction qui est à la base de l'holographie. Nous utiliserons comme exemple un élément bien connu en optique diffractive: *la grille*.

Une grille diffractive, comme son nom l'indique, est un agencement planaire de lignes parallèles équidistantes. Si la grille est reproduite sur un support transparent (une diapositive, par exemple) ces lignes pourraient être les régions où la transmittance est nulle (bandes opaques) (voir figure 1.3a). Il n'est pas nécessaire que les transitions opaques-transparentes soient brusques; elles peuvent être graduelles (équation 1.16).

Si cette grille est éclairée par un faisceau cohérent perpendiculaire au plan de la grille, une partie du faisceau sera bloquée par les bandes opaques (transmittance nulle) alors que l'autre sera transmise. Le principe de Huygens [37] nous dit qu'en tout moment et quelle que soit sa position, chaque point de l'espace peut être considéré comme une source radiale (qui émet dans toutes les directions). Donc, en utilisant les conventions de la figure 1.3a, si les bandes transparentes sont suffisamment minces, il est possible de modéliser selon l'axe y le faisceau de lumière au plan de la grille comme une colonne verticale de sources lumineuses distinctes qui émettent dans toutes les directions (figure 1.3b).

En considérant deux sources en particulier (figure 1.3c), on remarque que dans certaines directions il y aura *interférence constructive* (régions claires où les sommets et les

P153385.PDF [Page: 35 of 191]

creux s'additionnent), alors que dans d'autres, il y aura *interférence destructive* (régions sombres où les sommets d'additionnent aux creux et vice-versa). En fait, l'interférence constructive aura lieu précisément lorsque la différence du trajet optique (i.e. le trajet parcouru par un faisceau) entre deux faisceaux sera un multiple entier de la longueur d'onde de la lumière utilisée. Si d est la distance de séparation entre deux sources, il y aura donc interférence constructive (voir figure 1.3c) lorsque

$$d\sin\theta = n\lambda\,. \tag{1.17}$$

Cette condition est connue sous le nom de *condition de Bragg* [24]. En résumé, la grille a donc divisé un faisceau incident perpendiculaire en plusieurs faisceaux émis dans diverses directions dictées par la condition de Bragg.

1.2.4 Holographie

C'est à Denis Gabor que l'on doit l'invention de l'holographie [24][37]. Initialement, Gabor proposa l'holographie comme technique d'imagerie sans lentilles. Aujourd'hui,

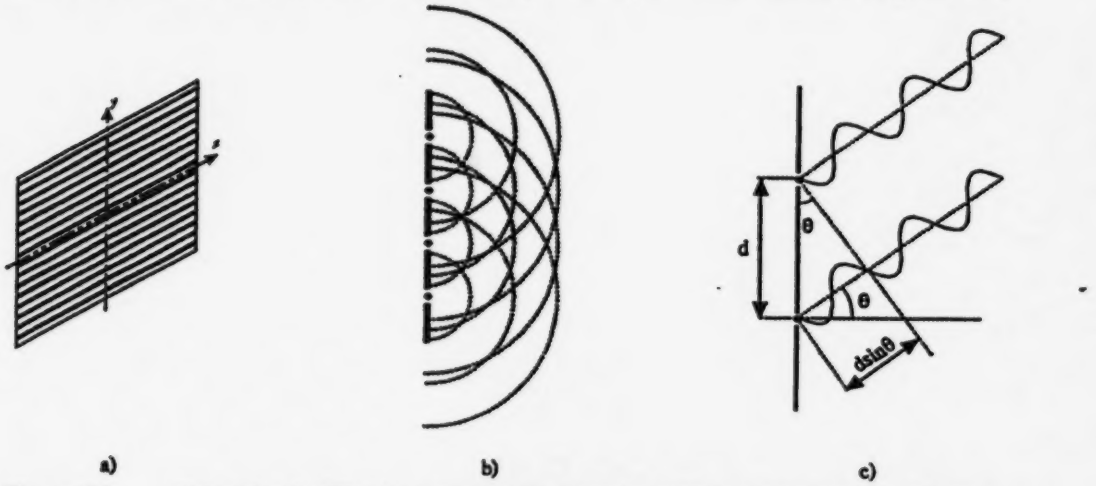


Figure 1.3: Phénomène de diffraction pour une grille diffractive. a) Grille diffractive; b) Modélisation par une colonne de sources radiales; c) Condition d'interférence constructive.

P153385.PDF [Page: 36 of 191]

l'holographie est utilisée dans une foule de domaines dont le traitement optique de l'information.

L'holographie conventionnelle est l'enregistrement (sur un médium photosensible) du patron d'interférence entre deux faisceaux de lumière cohérente. Un des faisceaux est appelé *faisceau objet*, tandis que l'autre porte le nom de *faisceau de référence*. Dénotons par O le faisceau objet et par R le faisceau référence. Ces faisceaux sont tous deux décrits par des fonctions complexes en raison de leur nature cohérente (i.e. information sur la phase et sur l'amplitude). Si on laisse interférer ces deux faisceaux et qu'on enregistre le patron d'interférence, l'intensité de ce patron sera donnée par

$$|O+R|^2 = |O|^2 + |R|^2 + OR^* + O^*R\,. \tag{1.18}$$

Lorsque nous développons la pellicule, la transmittance du filtre résultant est proportionnelle à l'intensité du patron d'interférence (si nous demeurons dans la région linéaire du film). Ainsi, la transmittance du film est donnée par

$$\beta|O+R|^2 = \beta|O|^2 + \beta|R|^2 + \beta OR^* + \beta O^*R\,, \tag{1.19}$$

où β est la constante de proportionnalité du film.

Si on éclaire à présent ce filtre uniquement avec le faisceau de référence, le patron d'intensité sur un plan immédiatement après le filtre sera

$$\beta\left(|O+R|^2\right)R = \beta|O|^2R + \beta|R|^2R + \beta O^*RR + \beta OR^*R \tag{1.20}$$

$$= \beta\left(|O|^2 + |R|^2\right)R + \beta R^2O^* + \alpha O\,, \tag{1.21}$$

où $\alpha = \beta RR^*$. Si R est une onde plane, $RR^* = |R|^2 \Rightarrow \alpha = \beta|R|^2 =$ scalaire. Si nous examinons à présent les trois termes produits, nous constatons que le premier est une onde plane (R), dont l'amplitude est modulée par le terme scalaire $\beta\left(|O|^2 + |R|^2\right)$, le deuxième est le conjugué du faisceau objet modulé par βR^2, une valeur complexe, et le troisième

P153385.PDF [Page: 37 of 191]

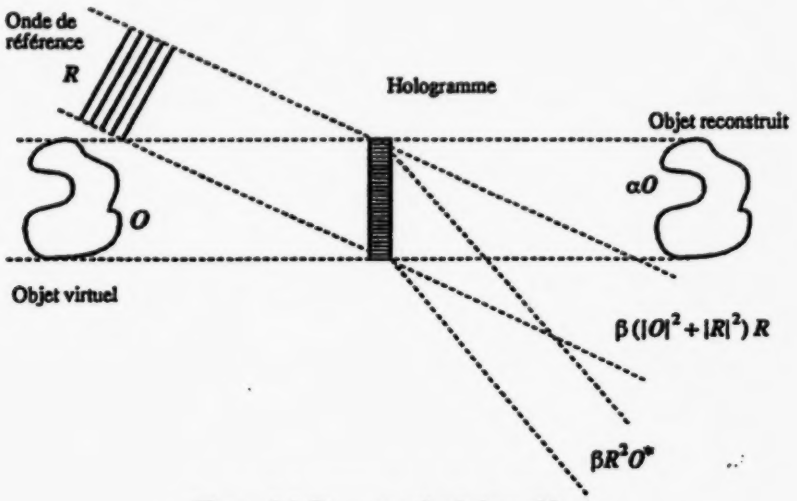


Figure 1.4: Reconstruction holographique.

terme est le faisceau objet (atténué par le scalaire α). Nous avons donc réussi à reproduire le faisceau objet même en l'absence de l'objet.

Pour être en mesure d'observer uniquement le faisceau objet (et non les deux autres faisceaux produits), il faut séparer spatialement les trois faisceaux (appelés *ordres*). Cette séparation se fait en variant l'angle d'incidence de l'onde plane de référence, tel qu'illustré à la figure 1.4 (voir [31] pour plus de détails sur la technique de séparation).

1.3 Éléments de théorie en électronique

Dans cette section, nous révisons les notions théoriques importantes en électronique. Notre survol commence par un rappel du principe de fonctionnement du transistor MOSFET. Ses divers modes d'utilisation sont par la suite couverts. Finalement, quelques circuits de base sont analysés.

1.3.1 Le transistor à effet de champ MOSFET

Le transistor à effet de champ est le composant vital de la grande majorité des circuits intégrés de nos jours. Il existe plusieurs types de transistors à effet de champ: JFET, MES-

FET, MOSFET, etc.). Le type qui nous intéresse ici est le MOSFET, plus précisément le MOSFET à enrichissement. Ce dernier est constitué de quatre ports: la grille (G) le drain (D), la source (S) et le substrat (B). Le substrat est relié à la source ou à un des potentiels d'alimentation d'un circuit intégré; le transistor devient alors un composant à trois ports. Il existe deux types de transistors MOSFET: les MOSFET à canal *n* (appelés nMOS) et les MOSFET à canal *p* (appelés pMOS). Sous le drain et la source d'un transistor nMOS se trouvent des régions de fort dopage *n*+. Ces deux régions sont séparées par un substrat dopé de type *p*. La région qui sépare le drain de la source porte le nom de canal. Au dessus du canal, séparée par une mince couche d'oxyde, se trouve la grille. Le MOSFET tire son nom de l'agencement des constituants au niveau de la grille: Metal-Oxide-Semiconductor Field Effect Transistor. Le transistor pMOS est de construction similaire à celle du nMOS, sauf que les régions de dopages *n*+ sont remplacées par des régions *p*+ et la région *p* par une région *n*.

Le principe d'opération est le suivant. Lorsqu'une tension positive est appliquée sur la grille, une accumulation d'électrons se produit sous l'oxyde de grille. Plus la tension est élevée, plus l'accumulation sera grande. Ainsi, lorsque la tension drain-source V_{ds} est élevée, un canal de conduction se forme alors entre le drain et la source. Les porteurs parcourent le canal de la source vers le drain. C'est pourquoi on parle de MOSFET *à canal n* car les porteurs (les électrons) sont de charge négative. Comme les porteurs sont des électrons, le courant circule donc en sens inverse des électrons, soit du drain vers la source.

Pour une tension de grille V_{gs} donnée, le courant I_{ds} croit d'abord linéairement avec la tension V_{ds}, selon la loi d'Ohm. Le transistor agit alors comme une résistance. Cependant, au fur et à mesure que la tension V_{ds} augmente, la tension V_{gc} de la grille au canal diminue. Le canal commence alors à se rétrécir et ce rétrécissement est plus important au niveau du drain. Lorsque V_{gd} diminue en dessous d'un certain seuil, V_{th}, caractéristique

P153385.PDF [Page: 39 of 191]

du transistor, le canal est *étranglé* et le courant cesse d'augmenter avec la tension V_{ds}. On dira alors que le transistor est en *saturation*. Noter que pour qu'un courant I_{ds} substantiel puisse circuler dans le canal, il faut que $V_{gs} > V_{th}$.

La tension V_{ds} à laquelle se produit la saturation dépend donc du seuil du transistor V_{th} et de la tension V_{gs}. Mathématiquement, on démontre que la condition de saturation est obtenue si

$$V_{gd} < V_{th} \Rightarrow V_{ds} > V_{gs} - V_{th} . \tag{1.22}$$

Le modèle mathématique qui décrit le courant de drain I_{ds} est donné par l'équation suivante:

$$I_{ds} = \begin{cases} \beta (V_{gs} - V_{th} - \frac{V_{ds}}{2}) V_{ds} & \text{si } V_{ds} < V_{gs} - V_{th} \\ \frac{\beta}{2} (V_{gs} - V_{th})^2 & \text{si } V_{ds} \geq V_{gs} - V_{th} \\ 0 & \text{si } V_{gs} \leq V_{th} \end{cases} \tag{1.23}$$

où β est un paramètre qui dépend de la géométrie du transistor et qui a pour unités des A/V^2.

Le modèle (1.23) est un modèle très simplifié qui ne tient pas compte de deux comportements importants: l'effet Early et la conduction infraseuil. L'effet Early permet d'expliquer le rétrécissement de la longueur du canal dû à l'élargissement des zones $n+$ sous le drain et la source. Comme le canal est moins long, sa résistance diminue et donc le courant I_{ds} augmente. Ce comportement se traduit par un courant de saturation qui croit légèrement de façon linéaire avec V_{ds}. Autrement dit, en saturation, la *conductance* du transistor, définie comme le taux de variation instantané du courant I_{ds} par rapport à la tension V_{ds} (i.e. $\partial I_{ds}/\partial V_{ds}$), est non-nulle. Pour tenir compte de l'effet Early, l'équation (1.23) doit être corrigée par le facteur $(1 + \lambda V_{ds})$, où λ a pour unités des V^{-1}.

P153385.PDF [Page: 40 of 191]

La conduction infraseuil explique le comportement du transistor lorsque sa tension V_{gs} est inférieure au seuil V_{th} du transistor. Le modèle (1.23) considère un courant I_{ds} nul dans une telle situation. En fait, un très faible courant circule entre le drain et la source. De plus, ce courant croît exponentiellement avec la tension V_{gs}. L'expression mathématique décrivant le courant infraseuil est donnée par:

$$I_{ds} = I_0 e^{V_g/U_T}(e^{-V_s/U_T} - e^{-V_d/U_T}) , \tag{1.24}$$

$$I_{ds} = I_0 e^{V_{gs}/U_T}(1 - e^{-V_{ds}/U_T}) , \tag{1.25}$$

où I_0 est un courant qui dépend du procédé de fabrication et de la géométrie du transistor, U_T est le voltage thermique (i.e. $U_T = kT/q$). Lorsque V_{ds} est suffisamment élevé (environ quatre fois supérieur à U_T), le courant I_{ds} sature à une valeur constante qui dépend de V_{gs}, i.e. $I_{ds} = I_{sat} = I_0 e^{V_{gs}/U_T}$.

Deux des avantages de la conduction infraseuil est la faible consommation de puissance (environ 10^{-12} à 10^{-6} Watts pour un circuit typique) et le fait que le courant sature très rapidement ($V_{ds(sat)} \approx 4U_T$) [56].

1.3.2 Modes d'utilisation du transistor MOSFET

Selon la plage d'opération de ses diverses tensions de commande, le transistor peut être utilisé à de nombreuses fins. Dans les circuits numériques, par exemple, le transistor est essentiellement utilisé en tant qu'interrupteur de courant. Une tension de commande V_{gs} est appliquée à sa grille. Si $V_{gs} = 0$, aucun courant ne peut circuler et donc l'interrupteur est ouvert. Si $V_{gs} = V_{dd}$, la tension d'alimentation, l'interrupteur est fermé et un courant I_{ds} déterminé par l'équation (1.23) circule entre son drain et sa source. La plupart des portes logiques en CMOS sont constituées de transistors utilisés comme interrupteurs.

P153385.PDF [Page: 41 of 191]

En électronique analogique, les modes d'utilisation du transistor sont beaucoup plus nombreux. Par exemple, il peut être utilisé comme une source de courant[1] en l'opérant en saturation (i.e. à des tensions $V_{ds} > V_{gs} - V_{th}$). La tension de grille V_{gs} sert alors à ajuster la valeur du courant. Notez que l'effet Early éloigne le comportement du transistor du comportement d'une source de courant idéale. En effet, le courant I_{dssat} varie légèrement en fonction de la tension V_{ds}.

Le transistor peut également être utilisé en tant que résistance commandée. En effet, si ce dernier est opéré dans la région linéaire, i.e. $V_{ds} < V_{gs} - V_{th}$, le courant varie (presque) linéairement avec la tension V_{ds}, comme dans une résistance. La valeur de la résistance peut être commandée par la tension de grille. Plus la tension V_{ds} est faible, plus le courant est linéaire, comme le témoigne la première équation de (1.23).

Une application intéressante qui met bien en relief le potentiel d'un transistor MOS est la multiplication analogique. Si le transistor est opéré dans la région linéaire ($V_{ds} < V_{gs} - V_{th}$), le courant de drain I_{ds} est, pour des tensions V_{ds} faibles, proportionnel au produit des tensions V_{gs} et V_{ds}. En effet, dans cette région, le courant I_{ds} est donné par la première équation de (1.23):

$$I_{ds} = \beta (V_{gs} - V_{th} - \frac{V_{ds}}{2}) V_{ds}, \text{ avec } V_{ds} < V_{gs} - V_{th}. \qquad (1.26)$$

Si V_{ds} est petit, alors I_{ds} est proportionnel à $\beta (V_{gs} - V_{th}) V_{ds}$. Cette approche nous permet d'effectuer des multiplications uniquement dans le premier quadrant (i.e. les deux entrées et le produit sont positifs). Dans la section suivante, nous présentons, entre autres, un circuit capable d'effectuer une multiplication dans deux et quatre quadrants.

1. Les pMOS sont utilisés comme *sources* de courant alors que les nMOS sont utilisés comme *puits* de courant.

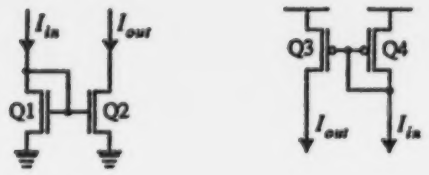


Figure 1.5: Miroirs de courant a) de type *n*; b) de type *p*.

1.3.3 Les circuits de base

Cette section décrit quelques circuits fondamentaux utilisés en électronique analogique. Ces circuits sont à la base des modules que nous présenterons à la section 3.3. Les circuits de base en électronique numérique ne seront pas discutés ici. Pour une excellente description, le lecteur est invité à consulter la référence [14].

Le premier circuit de base qui nous intéresse est le miroir de courant. Ce dernier tire son nom de sa fonction qui est de reproduire une image d'un courant. Les schémas de miroirs de courant sont illustrés à la figure 1.5.

Le transistor Q1 agit comme une diode puisque son drain est relié à sa grille. Sa tension V_{gs} sera donc ajustée de façon à ce que le courant I_{in} puisse circuler dans Q1. Cette même tension est également appliquée à la grille du transistor Q2. Si les deux transistors sont identiques, le même courant circulera dans Q2. Le principe est le même pour les miroirs de type pMOS.

Il n'est pas nécessaire que les deux transistors soient identiques, et dans certains cas, cette situation est même désirable. Par exemple, la largeur du transistor Q2 peut être doublée pour que le courant I_{out} soit le double de I_{in}.

Le problème majeur des miroirs de courant est que les copies sont rarement précises en raison des variations imprévisibles des paramètres des transistors (comme le degré de

P153385.PDF [Page: 43 of 191]

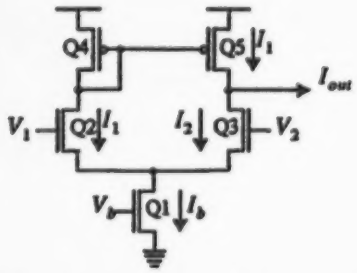


Figure 1.6: Amplificateur différentiel de transconductance.

dopage, par exemple). Ainsi, même si deux transistors sont voisins, il se peut fort bien que leur seuil soient différents, et par conséquent $I_{out} \neq I_{in}$.

Le deuxième circuit que nous décrivons est l'amplificateur différentiel de transconductance. Son schéma est donné à la figure 1.6. Un amplificateur de transconductance convertit une tension à un endroit du circuit en un courant *ailleurs* dans le circuit. L'amplificateur est dit *différentiel* puisqu'il amplifie non pas une tension mais une différence de tension entre deux de ses bornes. Le principe est simple: la tension V_b fixe un courant de polarisation I_b à travers Q1. Ce courant se répartit alors entre Q2 et Q3, selon les tensions V_1 et V_2. Le miroir de courant de type p agit ici comme un soustracteur de courant. Ce dernier reflète le courant I_1 dans sa branche de droite. En appliquant la loi de Kirchoff au nœud de sortie, on obtient $I_{out} = I_1 - I_2$. Si les deux branches sont parfaitement symétriques, le courant I_b se divisera également entre les deux branches et le courant de sortie sera nul. Si la tension V_1 est augmentée par rapport à V_2, une portion plus grande de I_b sera dirigée vers la branche de gauche, et donc $I_1 > I_2 \Rightarrow I_{out} > 0$. La situation inverse se produit si $V_2 > V_1$.

L'expression du courant de sortie dépend de l'intensité du courant de polarisation. Si le transistor Q1 opère en mode infraseuil, (i.e. courant de polarisation très faible), alors le courant de sortie est donné par ([56])

$$I_o = I_b \tanh\left(\frac{\kappa(V_1 - V_2)}{2U_T}\right), \tag{1.27}$$

où κ est une constante qui dépend du procédé de fabrication. Si, par contre, la tension de grille de Q1 est supérieure au seuil V_{th}, l'expression se complique quelque peu [46]

$$I_o = \frac{\beta}{2}(V_1 - V_2)\sqrt{\frac{4I_b}{\beta} - (V_1 - V_2)^2}\,. \tag{1.28}$$

Ces deux expressions assument que les transistors Q2 et Q3 opèrent dans la région de saturation et que le miroir de courant procure une reflectivité parfaite.

Nous avons démontré dans la section précédente comment réaliser un multiplieur analogique à l'aide d'un seul transistor. La précision d'un tel multiplieur est très faible en raison du comportement non-linéaire du courant de drain dans la région de triode. Il est possible, avec un circuit de 4 transistors, de réaliser un multiplieur qui annule l'effet non-linéaire et permet une multiplication dans les quatre quadrants. Le montage est illustré à la figure 1.7. Dans ce circuit, le courant de sortie I_o est donné par

$$I_o = I_{o1} - I_{o2} = I_1 + I_2 - (I_3 + I_4)\,. \tag{1.29}$$

Si les transistors opèrent tous en saturation, il est possible de démontrer que [46]

$$I_o = \beta(V_1 - V_3)(V_4 - V_2)\,. \tag{1.30}$$

L'avantage de ce circuit est sa linéarité accrue, son faible nombre de transistors, ainsi que sa grande plage de variation due au fait que les transistors opèrent tous en saturation.

P153385.PDF [Page: 45 of 191]

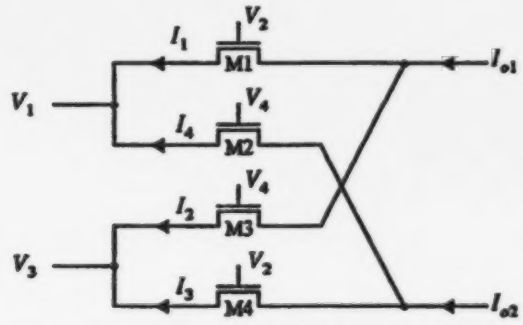


Figure 1.7: Multiplieur analogique à quatre quadrants.

Notre dernier circuit est le populaire multiplieur de Gilbert [56]. Son circuit est illustré à la figure 1.8. Il est constitué de trois paires différentielles nécessaires pour générer les quatre courants constituants de I_{out} (un courant pour chaque quadrant). Un miroir de courant effectue alors la différence entre les courants positifs (quadrants 1 et 2) et les courants négatifs (quadrants 3 et 4). L'expression du courant de sortie en fonction des tensions différentielles, lorsque Q1 opère en mode infraseuil, est donné par l'équation suivante [56]:

$$I_{out} = I_b \tanh\left(\frac{\kappa(V_1 - V_2)}{2U_T}\right) \tanh\left(\frac{\kappa(V_3 - V_4)}{2U_T}\right). \tag{1.31}$$

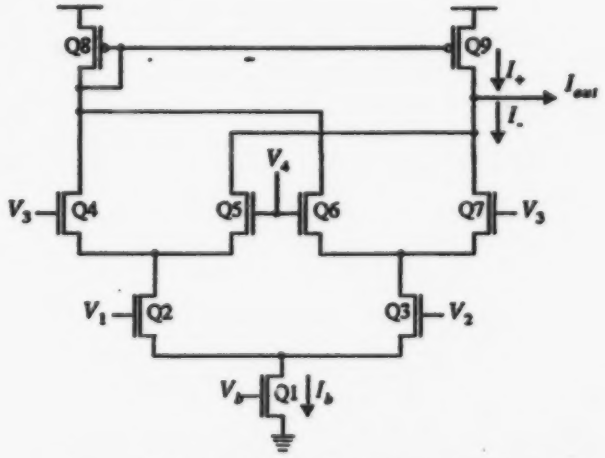


Figure 1.8: Multiplieur quatre quadrants de Gilbert.

P153385.PDF [Page: 46 of 191]

Pour des tensions différentielles faibles, $\tanh x \approx x$ et (1.31) devient

$$I_{out} \approx I_b \left(\frac{\kappa (V_1 - V_2)}{2U_T} \right) \left(\frac{\kappa (V_3 - V_4)}{2U_T} \right). \tag{1.32}$$

L'avantage du circuit de la figure 1.8 par rapport à celui de la figure 1.7 est que le courant de sortie est fourni par l'alimentation, alors que dans l'autre circuit, les modules connectés aux tensions V_1 et V_3 doivent être en mesure de fournir le courant de sortie: les entrées ne sont donc pas isolées de la sortie.

Dans ce chapitre, nous avons résumé les principaux éléments de théorie en réseaux de neurones artificiels, en optique et en électronique. Cette révision était nécessaire pour mieux comprendre les architectures que nous allons décrire dans les chapitre II et III. Le chapitre suivant traite des principales architectures optiques pour la réalisation des réseaux de neurones.

CHAPITRE II RÉSEAUX DE NEURONES OPTIQUES

Dans ce chapitre, nous allons tenter de décrire les plus importantes architectures de réseaux de neurones optiques. Le terme "optique" englobe aussi bien les architectures entièrement optiques que les architectures hybrides faisant appel soit à l'électro-optique, soit à l'acousto-optique, soit aux deux. Dans un effort de synthèse, nous avons choisi de regrouper les architectures selon trois grandes catégories. Ces catégories sont:

- les architectures à processeurs algébriques optiques;
- les architectures à corrélateurs optiques;
- les architectures à interconnexions holographiques.

Une section décrira en détail chacune de ces catégories. Pour chaque architecture, nous discuterons des considérations d'ordre pratique découlant de leur mise en œuvre ainsi que des performances escomptées ou mesurées, selon le cas. Il est important de mentionner qu'étant donné la faible maturité de la technologie des RNA optiques, la plupart des architectures qui seront décrites n'existent que sous forme de prototypes. Or, les performances d'un prototype sont souvent bien loin des performances du produit fini. Nous tenterons donc, si cela est possible, d'ajuster les performances en conséquence pour permettre une comparaison plus équitable avec la technologie électronique.

P153385.PDF [Page: 48 of 191]

2.1 Composants optiques

Dans ce chapitre, nous allons décrire brièvement les principaux composants utilisés en traitement optique de l'information, plus particulièrement en réseaux de neurones optiques. Ces composants sont les lentilles, les émetteurs, les détecteurs, les modulateurs spatiaux de lumière et les hologrammes.

2.1.1 Lentilles

La lentille est sans conteste le composant le plus utilisé en optique. Une lentille est généralement réalisée sous la forme d'une plaque de verre usiné dont l'épaisseur varie en fonction de la position. Lorsque l'épaisseur varie radialement, la lentille est dite *sphérique* tandis que lorsque l'épaisseur varie selon une seule direction, la lentille est dite *cylindrique*. La figure 2.1 illustre deux exemples de lentilles. L'épaisseur variable de la lentille a pour effet de modifier la phase des faisceaux incidents en fonction de la position, et ce, dans le but de les faire dévier.

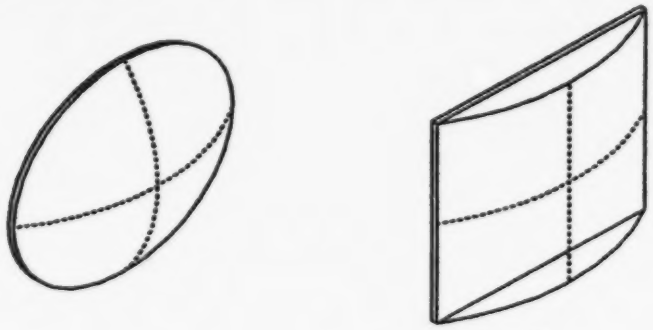


Figure 2.1: Exemple de lentilles. a) Sphérique; b) Cylindrique.

De nombreuses applications existent pour les lentilles, qu'elles soient sphériques ou cylindriques. Par exemple, une application bien connue, la photographie, utilise une ou plusieurs lentilles pour imager une scène sur une pellicule photographique. En traitement

P153385.PDF [Page: 49 of 191]

optique de l'information, les lentilles sont souvent utilisées pour leur propriété de transformation de Fourier. En effet, si un patron lumineux $i(x, y)$ se trouve au plan focal arrière d'une lentille convexe (voir plan xy arrière, figure 2.8), le patron résultant au plan focal avant (voir plan uv, figure 2.8) est la transformée de Fourier 2-D du patron d'entrée, soit $I(u, v)$, multipliée par une constante complexe [31][1]. Les coordonnées fréquentielles u et v sont reliées aux coordonnées spatiales du plan focal avant x' et y' par les relations

$$u = \frac{x'}{\lambda f} \text{ et } v = \frac{y'}{\lambda f}. \tag{2.1}$$

où λ est la longueur d'onde de la lumière monochromatique utilisée et f est la distance focale de la lentille. La constante qui multiplie la transformée de Fourier est $1/(j\lambda f)$, de sorte que la transformation qu'effectue la lentille sur le patron d'entrée est

$$o(x', y') = \frac{1}{j\lambda f}\int_{-\infty}^{\infty}\int i(x, y)\, e^{-j\frac{2\pi}{\lambda f}(xx'+yy')}\, dxdy = \frac{1}{j\lambda f} I\left(\frac{x'}{\lambda f}, \frac{y'}{\lambda f}\right) = \frac{1}{j\lambda f} I(u, v). \tag{2.2}$$

La présence du facteur multiplicatif devant la transformée de Fourier peut être expliqué par le principe de la conservation de l'énergie. En effet, l'énergie totale du patron d'entrée devra être égale à l'énergie totale du patron produit au plan de Fourier, en supposant que toute l'énergie incidente traverse la lentille. Mathématiquement,

$$\int_{-\infty}^{\infty}\int |i(x, y)|^2 dxdy = \int_{-\infty}^{\infty}\int |o(x', y')|^2 dx'dy'. \tag{2.3}$$

$$\text{Or, } \int_{-\infty}^{\infty}\int |o(x', y')|^2 dx'dy' = \int_{-\infty}^{\infty}\int \left|\frac{1}{j\lambda f} I\left(\frac{x'}{\lambda f}, \frac{y'}{\lambda f}\right)\right|^2 dx'dy'. \tag{2.4}$$

1. La transformée de Fourier 2-D est définie par $I(u, v) = \int\int_{-\infty}^{\infty} i(x, y)\, e^{-j2\pi(xu+yv)}\, dxdy$

P153385.PDF [Page: 50 of 191]

En utilisant le changement de variable (2.1), (2.4) devient

$$\int_{-\infty}^{\infty}\int\left|\frac{1}{j\lambda f}I\left(\frac{x'}{\lambda f},\frac{y'}{\lambda f}\right)\right|^2 dx'dy' = \frac{1}{(\lambda f)^2}\int_{-\infty}^{\infty}\int|I(u,v)|^2(\lambda f)^2 dudv = \int_{-\infty}^{\infty}\int|I(u,v)|^2 dudv \quad (2.5)$$

Or, d'après le théorème de Parseval,

$$\int_{-\infty}^{\infty}\int|I(u,v)|^2 dudv = \int_{-\infty}^{\infty}\int|i(x,y)|^2 dxdy, \quad (2.6)$$

ce qui complète la preuve.

2.1.2 Émetteurs

Tout comme en électronique où l'information est représentée physiquement par la modulation du courant dans un fil ou d'une tension à un nœud, en optique, l'information est représentée physiquement par la modulation de l'amplitude ou de la phase d'un faisceau lumineux[1]. Bien souvent, cette information est déjà disponible sous forme électronique et il est donc nécessaire de la convertir de l'électronique à l'optique. Un composant capable d'accomplir cette tâche porte le nom d'*émetteur* ou de *source lumineuse*.

Il existe deux principaux types d'émetteurs couramment utilisés en traitement optique de l'information: les LED et les lasers. Une LED (de l'anglais Light Emitting Diode) convertit un signal électrique en lumière incohérente. Cette conversion, contrairement à l'ampoule électrique domestique, est très efficace en terme d'énergie (i.e. environ 45% de l'énergie électrique incidente est convertie en énergie optique [24]). La LED est utile pour les applications où la cohérence spatiale n'est pas requise.

1. Il est également possible de moduler la polarisation du faisceau si ce dernier est cohérent.

P153385.PDF [Page: 51 of 191]

Lorsque la cohérence spatiale est requise, les lasers sont plutôt utilisés. Un laser émet un faisceau cohérent monochromatique (une seule longueur d'onde) très directionnel et très intense. Le nom laser vient de l'acronyme anglais (Light Amplification by Stimulated Emission of Radiation).

Deux types principaux de lasers existent: les diodes lasers et les lasers à gaz. Les lasers à gaz sont en général plus gros et plus puissants que les diodes lasers, mais leur efficacité est beaucoup moindre (inférieure à 10% pour les lasers à gaz [24] et de l'ordre de 20% pour les diodes laser [55]).

Bien souvent, l'information n'est pas introduite dans le système optique en modulant directement l'intensité de plusieurs sources (i.e. une source par signal); une seule source est plutôt utilisée. Son faisceau est d'abord élargi et collimaté (rendu parallèle) par un système de lentilles. Le large faisceau est alors modulé par un modulateur spatial de lumière (MSL)(voir section 2.1.4 en page 30). L'utilisation d'un seul laser est avantageuse, car bien souvent il est difficile d'obtenir deux sources cohérentes dont la longueur d'onde est identique. De plus, la puissance accrue du laser à gaz peut ainsi être mise à profit.

2.1.3 Détecteurs

À l'inverse de l'émetteur, le détecteur convertit un faisceau lumineux en un signal électrique. Des exemples de détecteurs sont les photodiodes PIN et avalanche, les photoconducteurs, les photomultiplieurs et les CCD (pour une desciption de ces divers détecteurs, consulter la référence [24]).

On retrouve les détecteurs dans les systèmes opto-électroniques, là ou les conversions optique/électronique sont nécessaires. Même dans les systèmes entièrement optiques, on retrouvera souvent des détecteurs à la sortie du système pour capter ou enregistrer l'image de sortie (par l'entremise d'une caméra CCD, par exemple).

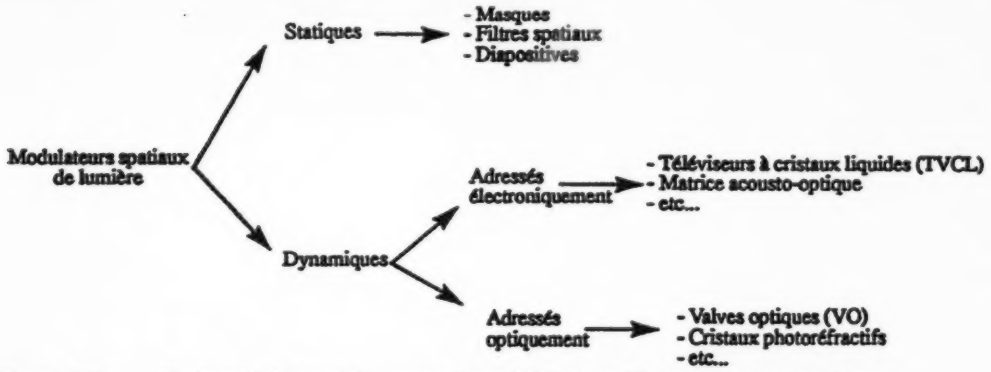


Figure 2.2: Taxinomie des modulateurs spatiaux de lumière utilisés en traitement optique de l'information.

Une faiblesse importante de tous les détecteurs est qu'ils sont capables de détecter uniquement l'intensité lumineuse[1]; toute information sur la phase du faisceau lumineux est perdue après détection.

2.1.4 Modulateurs spatiaux de lumière

L'outil le plus important en traitement optique de l'information est probablement le *modulateur spatial de lumière* (MSL). Un MSL a pour but de modifier (moduler) la distribution lumineuse de la section d'un faisceau incident (d'où le nom modulateur *spatial* de lumière). Un exemple simple de MSL est la diapositive. Lorsqu'un faisceau de lumière blanche traverse une diapositive, le faisceau résultant porte l'information spatiale et chromatique de l'image qui se trouvait sur la diapositive. La diapositive a donc *modulé* le faisceau incident avec l'information qu'elle contenait.

Il existe une multitude de modulateurs spatiaux de lumière. La figure 2.2 illustre une taxonomie possible. Les MSL sont d'abord divisés en deux grandes classes: statiques et dynamiques. Un MSL statique modulera un faisceau incident toujours de la même façon. La diapositive est un exemple de MSL statique. Les MSL dynamiques, quant à eux, modu-

1. Il existe des détecteurs cohérents mais ceux-ci sont coûteux et difficile à réaliser en pratique.

P153385.PDF [Page: 53 of 191]

lent le faisceau incident de façon dynamique (i.e. contrôlable). Selon le type d'adressage, on distingue les MSL adressés *électroniquement* ou *optiquement*. Un exemple de MSL adressé électroniquement est le téléviseur à cristaux liquides (TVCL) que l'on retrouve sur la plupart des écrans d'ordinateurs portatifs. Un MSL adressé optiquement module le faisceau incident avec un autre faisceau (souvent appelé faisceau d'écriture). Un exemple est la valve optique. La fonction d'une valve optique est de reproduire la distribution de lumière présente à son entrée (le faisceau d'écriture) en lumière cohérente à sa sortie, multipliée par un gain. Le fait que le faisceau de sortie soit cohérent permet d'utiliser la valve comme transducteur d'entrée dans un système optique cohérent. Ainsi, l'image d'entrée, possiblement incohérente, deviendrait cohérente à la sortie de la valve optique.

Nous aurons l'occasion de décrire plus en détail certains MSL lorsque nous aborderons le sujet des réseaux de neurones optiques.

2.1.5 Hologrammes

Comme nous l'avons vu à la section 1.2.4, un hologramme est l'enregistrement du patron d'interférence entre deux faisceaux de lumière cohérente. Bien souvent, cet enregistrement se fait de façon optique (en enregistrant le patron sur une pellicule photosensible), mais cette technique n'est pas unique. Une technique de plus en plus populaire est la génération d'hologrammes par ordinateur (GHO)[24], [55]. Avec cette technique, il est même possible de générer des hologrammes qui autrement n'auraient pas pu être enregistrés optiquement.

Il existe deux grandes classes d'hologrammes: les hologrammes *d'amplitude* et les hologrammes *de phase*. Un hologramme d'amplitude agit uniquement sur l'amplitude du faisceau cohérent incident. Un exemple d'hologramme d'amplitude est la grille diffractive. Un hologramme de phase agit uniquement sur la phase du faisceau incident. Un holo-

P153385.PDF [Page: 54 of 191]

gramme de phase bien connu est le réseau binaire de Dammann, qui transforme une source lumineuse à l'entrée en une matrice de sources lumineuses d'intensités égales à la sortie. Les cristaux photoréfractifs peuvent également être utilisés comme hologrammes de phase. Un cristal photoréfractif est une structure cristalline dont l'indice de réfraction peut être modifié localement par une interférence constructive de faisceaux lumineux.

L'hologramme de phase possède un avantage marqué par rapport à l'hologramme d'amplitude: toute l'énergie incidente à un hologramme de phase est transmise. Pour un hologramme d'amplitude, une portion de l'énergie incidente est absorbée par l'hologramme, contribuant ainsi à décroître l'efficacité diffractionnelle.

2.2 Architectures à processeurs algébriques optiques

2.2.1 Principe

Dans la section 1.1, nous avons décrit le principe d'opération d'un réseau de neurone sans rétroaction. L'équation (1.4), qui exprime le calcul des entrants nets, est une équation algébrique linéaire. Elle représente une importante partie des calculs que doit effectuer une couche de neurones à chaque itération, le reste de l'effort de calcul étant dédié au seuillage. Une architecture optique qui réalise l'équation (1.4) porte le nom de *processeur algébrique optique*. Ce processeur peut donc aisément constituer le noyau d'une architecture neuronale. Le reste de cette section décrira en détail le fonctionnement d'un processeur algébrique optique bien utile: le *multiplieur vecteur-matrice* (MVM). Nous décrirons par la suite comment ce processeur peut être intégré dans une architecture neuronale.

2.2.2 Le multiplieur vecteur-matrice optique

La figure 2.3 illustre le montage d'un multiplieur vecteur-matrice optique. Ce montage se décrit de gauche à droite. À l'extrême gauche se trouve une rangée d'émetteurs dont la fonction est de reproduire optiquement le vecteur d'entrée $\mathbf{X}$. Les émetteurs sont soit des

P153385.PDF [Page: 55 of 191]

LED (dans une architecture à lumière incohérente), soit des diodes laser (dans une architecture à lumière cohérente). Il y a autant d'émetteurs que d'éléments dans le vecteur **X**, soit N. Le but de la première lentille cylindrique est d'étaler sur chacune des colonnes de la matrice les N faisceaux lumineux émis par la rangée d'émetteurs. La lentille agit donc comme un collimateur vertical. Les N colonnes de lumière sont ensuite modulées par un MSL de $N \times M$ pixels placé directement devant la lentille. Le type de MSL utilisé est souvent le téléviseur à cristaux liquides (TVCL), mais on peut également utiliser des matrices acousto-optiques. Les valeurs des pixels du MSL correspondent aux valeurs des éléments de la matrice de poids **W**. Le rôle de la seconde lentille cylindrique est de faire converger horizontalement les faisceaux transmis par le MSL. Une colonne de détecteurs est ensuite placée au foyer de cette lentille pour receuillir les N produits scalaires en détectant l'intensité lumineuse en chaque point sur la colonne. De cette façon, N produits scalaires sont calculés en parallèle.

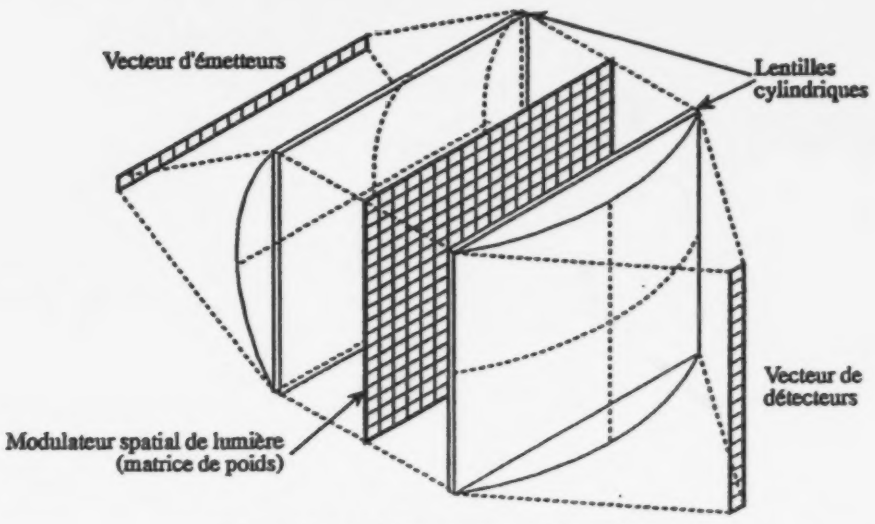


Figure 2.3: Multiplieur vecteur-matrice optique.

P153385.PDF [Page: 56 of 191]

Pour obtenir une architecture neuronale à partir de ce processeur algébrique, il suffit d'appliquer aux signaux de sortie des détecteurs une non-linéarité comme celle de la figure 1.1b. Cette opération est souvent réalisée électroniquement, ce qui nécessite M conversions optique/électronique au niveau des détecteurs. Selon le paradigme neuronal utilisé, une rétroaction peut être introduite en reportant à l'entrée les signaux de sortie pour obtenir un réseau de Hopfield, par exemple (voir figure 2.4a), ou plusieurs modules MVM peuvent être mis en cascade pour obtenir un réseau de perceptrons multicouches (voir figure 2.4b).

2.2.3 Considérations d'ordre pratique

La plupart des téléviseurs à cristaux liquides sont adressés électroniquement par un réseau de lignes perpendiculaires. Pour adresser un TVCL $N \times N$, il faut $2N$ lignes d'adresses. Ceci a pour conséquence de limiter la vitesse à laquelle la matrice sera entièrement mise à jour puisque les pixels ne sont pas tous adressés en même temps. Pour permettre un adressage totalement parallèle, il faut un total de N^2 lignes de contrôle. Un réseau $N \times N$ adressé par N^2 lignes de contrôle occupe une surface de l'ordre de N^3. Plus N est grand, plus l'espace entre les pixels devra être grand pour maintenir une matrice

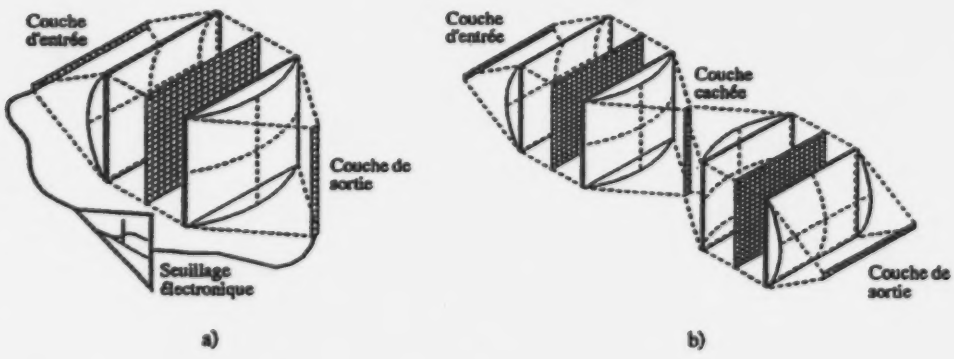


Figure 2.4: Réseaux de neurones à noyau algébrique optique. a) Avec rétroaction (ex. Hopfield); b) Perceptron multicouche (ex. backpropagation).

P153385.PDF [Page: 57 of 191]

régulière. Pour une valeur de *N* élevée (de l'ordre de 256), cet espace peut devenir prohibitif pour certaines applications en raison de la faible résolution spatiale résultante (en pl/mm). C'est pourquoi la majorité des TVCL sont adressés électroniquement par deux ensembles de lignes perpendiculaires. Une conséquence directe de cette contrainte est la nécessité d'introduire de la synchronisation dans le système. En effet, comme tous les pixels ne sont pas mis à jour en même temps, il faut s'assurer que les détecteurs soient actifs durant la période d'activation des pixels qu'ils doivent intégrer, sans quoi les calculs riqueraient de se chevaucher. Certains manufacturiers de TVCL ont éliminé ce problème en concevant des pixels avec mémoire. Ces derniers, une fois mis à jour, conserveront leur état jusqu'au prochain adressage. Ainsi, il suffit de mettre à jour la matrice au complet pour ensuite activer tous les détecteurs en parallèle.

L'adressage électronique a l'avantage de procurer au système une certaine versatilité. En effet, un TVCL adressé électroniquement peut être aisément programmé par un contrôleur externe capable d'effectuer un prétraitement utile sur les données avant de les afficher. Le premier exemple de prétraitement qui vient à l'esprit est l'ajustement des poids d'interconnexion lors de l'apprentissage dans les architectures neuronales. L'algorithme d'apprentissage peut alors bénéficier de la précision accrue des processeurs électroniques digitaux. Deux autres applications sont l'augmentation de la précision des calculs optiques et le support des nombres négatifs, que nous décrivons ici brièvement.

L'augmentation de la précision peut être intéressante lorsque l'application exige un niveau de précision élevé. Un processeur algébrique optique sans technique d'augmentation de précision est capable, en général, de supporter au plus 8 bits de précision [24], [28]. Cette limite est imposée par une combinaison du nombre limité de niveaux de gris disponibles sur le MSL, du bruit de transmission et des interférences. Divers algorithmes existent pour augmenter la précision (en échange d'un débit de calcul réduit). Mention-

P153385.PDF [Page: 58 of 191]

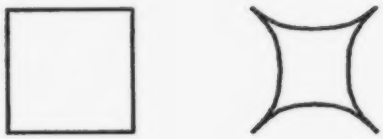
nons entre autres la technique de multiplication numérique par convolution analogique (MNCA), le partionnement numérique, le système numérique redondant et le système numérique résiduel (ces techniques sont toutes décrites dans les références [28] et [66]). Peu importe la technique utilisée, les techniques de contrôle sont en général complexes et requièrent l'assistance de l'électronique. Il est à noter que si une précision de calcul trop élevée est requise, il est possible qu'un processeur entièrement électronique soit plus performant qu'un processeur optique [28]. C'est ainsi que, bien qu'il soit possible d'augmenter la précision des processeurs algébriques optiques, les véritables gains de performances comparativement à l'électronique s'obtiennent en mode de précision réduite (i.e. sans technique d'augmentation de la précision).

Un problème avec le processeur algébrique optique est le support des nombres négatifs. En effet, en optique, l'information est communiquée par la modulation de l'intensité d'un faisceau lumineux. Plus le faisceau est intense, plus le nombre est élevé et vice-versa. Cette manière de représenter l'information pose un problème lorsque vient le temps de représenter des valeurs négatives. En effet, une intensité ne peut avoir de valeur négative. Une technique couramment utilisée dans les processeurs algébriques optiques consiste à séparer les opérations positives des opérations négatives et effectuer la soustraction électroniquement. Par exemple, l'opération suivante

$$[2 \times 3] + [4 \times (-5)] = -14, \tag{2.7}$$

peut être effectuée en deux temps. Dans un premier temps, toutes les valeurs positives sont affichées sur la matrice à cristaux liquides (CL), les valeurs négatives étant forçées à zéro. Un premier résultat intermédiaire est alors calculé. Dans un deuxième temps, la valeur absolue des valeurs négatives est affichée, les valeurs positives étant forçées à zéro. Un deuxième résultat intermédiaire est ainsi calculé. Par après, les deux résultats intermédiai-

P153385.PDF [Page: 59 of 191]



a) b)

Figure 2.5: Exemple d'aberration sphérique; a) image d'entrée; b) image de sortie.

res (tous deux positifs) sont détectés, convertis sous forme numérique, puis soustraits électroniquement. Ainsi, l'équation (2.7) est décomposée ainsi:

$$[2\times3+4\times0]-[2\times0+4\times5]=6-20=-14. \qquad (2.8)$$

Les processeurs algébriques optiques souffrent d'importants problèmes d'alignement. En effet, pour une matrice de taille moyenne (de l'ordre de 256x256), l'étalement uniforme du vecteur d'entrée sur la matrice du TVCL devient très difficile. Une des raisons est l'écart du comportement idéal des lentilles cylindriques. Les lentilles réelles souffrent bien souvent de problèmes d'aberration. Une aberration est une distorsion non-linéaire produite par la lentille due à sa forme. Un exemple d'aberration sphérique est illustré à la figure 2.5.

Outre les problèmes d'alignement, les processeurs algébriques optiques souffrent également du problème de *vignetage* des lentilles. Le vignetage est une conséquence de l'ouverture finie d'une lentille. Le principe est que plus une source lumineuse est proche de l'axe optique de la lentille, plus la quantité d'énergie captée par la lentille sera grande. Plus cette source s'éloigne de l'axe, moins la quantité d'énergie sera grande. Les pixels situés loin de l'axe optique seront défavorisés par rapport aux autres pixels. Comme l'information est représentée par l'intensité d'un faisceau lumineux, ce phénomène peut introduire des erreurs.

2.2.4 Performances

Deux facteurs limitent la performance dans les architectures neuronales à noyau algébrique optique. Le premier facteur est propre à la technologie TVCL. Pour cette technologie, les deux principales limitations sont la résolution (i.e. le nombre de pixels) et le temps de réponse (la vitesse de mise à jour des pixels). Le deuxième facteur limitant est attribué aux conversions optique/électronique et électronique/optique. Comme la plupart du temps, les non-linéarités sont réalisées électroniquement, il est nécessaire de convertir les signaux optiques en signaux électroniques. Pour réintroduire les résultats dans le système optique, une autre conversion, cette fois électronique/optique, est nécessaire. Ces deux opérations sont coûteuses en temps d'exécution. Dans ce qui suit, nous allons tenter de quantifier les performances qu'il est possible d'obtenir avec ce genre d'architectures.

Le tableau 2.1 énumère quelques caractéristiques importantes de téléviseurs à cristaux liquides. La colonne «Nombre de neurones» mérite quelques précisions. Rappelons tout d'abord qu'un multiplieur vecteur-matrice a pour but de calculer l'entrant net d'un ensemble de neurones (voir équation (1.4)). Le nombre de neurones simulables avec ce genre d'architectures dépend de la topologie d'interconnexion. Par exemple, nous pourrions utiliser la matrice 320x220 du TVCL Kodak LC500 [49] pour interconnecter, soit 220 neurones d'entrée avec 320 neurones de sortie, soit 320 neurones d'entrée avec 220 neurones de sortie. Dans les deux cas, le nombre d'interconnexions demeure le même, soit $320 \times 200 = 64000$. Devons-nous dire que la TVCL permet de simuler 320 neurones (avec une entrance de 220) ou 220 neurones (avec une entrance de 320)? Pour résoudre l'ambiguïté et dans un effort de généralisation, nous allons dorénavant supposer que le nombre de neurones simulables par un MVM est la racine carré du nombre total de pixels disponibles. Donc, pour la Kodak LC500, le nombre de neurones simulables est $\sqrt{320 \times 200} \approx 250$.

P153385.PDF [Page: 61 of 191]

Tableau 2.1: Caractéristiques de quelques téléviseurs à cristaux liquides ([21]).

Type de TVCL	Type de cristaux liquides	Résolution (en pixels)	Taux maximum de rafraîchissement (en hertz)	Nombre de neurones (interconnexion N^2)	Connexions par seconde (cps)
Epson VPJ-2000 [23]	NT[a]	480x440	30	~460	$6{,}33 \times 10^5$
Kodak LCS00 [49]	NT	320x220	30	~250	$1{,}92 \times 10^6$
STC Ltd [47]	CLF[b]	128x128	165	128	$2{,}70 \times 10^6$
Displaytech DRO128A [20]	CLF	128x128	4000	128	$65{,}54 \times 10^6$
Displaytech DRO256A [20]	CLF	256x256	4000	256	$262{,}14 \times 10^6$
GEC-Marconi CMOS [30]	CLF	176x176	512	176	$15{,}86 \times 10^6$

a. NT: Nématique Torsadés.
b. CLF: Cristaux Liquides Ferro-électriques.

Ce genre d'architectures nous laisse malheureusement peu de choix quant à la topologie d'interconnexion. Elles sont particulièrement bien adaptées aux topologies où l'interconnexion entre les neurones est totale. En effet, à cause des contraintes matérielles imposées par les TVCL, les interconnexions doivent être spécifiées par une matrice $I \times J$ où I et J sont les nombres de neurones à l'entrée et à la sortie respectivement. Si on désire réduire l'entrance des neurones de sortie, il est nécessaire de forcer à zéro les entrées dans la matrice qui correspondent à des interconnexions non désirées. Il en résulte alors un gaspillage de la zone active de la TVCL.

Une autre contrainte est que les données à l'entrée doivent être disposées en une seule rangée (ou colonne), donc sur une seule dimension. Ceci pose un problème lorsque les données à traiter sont des images bidimensionnelles. Dans ce cas, il est nécessaire d'aligner les données sur une seule dimension («rasterize») avant de les présenter au MVM. La figure 2.6 illustre graphiquement les performances auxquelles on peut s'attendre avec des architectures neuronales à noyau algébrique optique utilisant les TVCL du tableau 2.1. Les droites montrent les performances lorsque la matrice n'est pas complètement utilisée. Il apparaît clairement que dans ce cas, le nombre de cps ne peut que baisser car la fré-

P153385.PDF [Page: 62 of 191]

quence de rafraîchissement demeure constante alors que le nombre de pixels (connexions) utilisés décroit. À titre de référence, nous avons également tracé les performances d'un multiplieur-accumulateur 8 bits TMC2208 de TRW [29]. Ce dernier, par sa nature séquentielle, procure un débit constant de 25Mcps[1], peu importe le nombre de neurones simulés. Le graphique montre que seules les TVCL de Displaytech peuvent espérer rivaliser avec le TMC2208 pour un nombre suffisamment élevé de neurones (>100). Il semble donc que l'utilisation de MVM optiques à TVCL ne soit pas justifiable pour des applications de réseaux de neurones: l'électronique offre généralement de meilleures performances à moindre coûts.

Il est important de mentionner que si l'application n'exige pas que la matrice soit mise à jour périodiquement (comme c'est le cas pour des réseaux de neurones pour lesquels l'apprentissage est terminé), les performances statiques peuvent dépasser les performan-

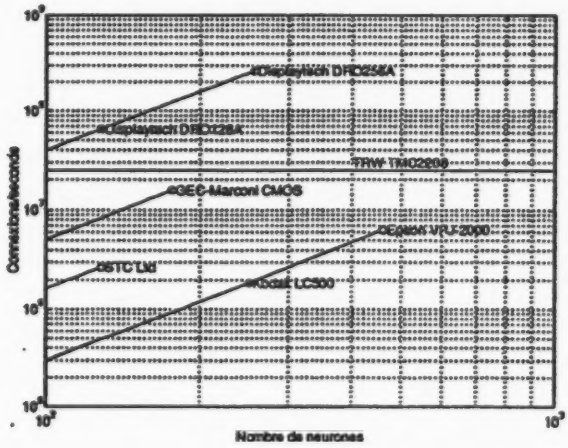


Figure 2.6: Performances pour architectures à processeur algébrique optique utilisant des TVCL.

1. Pour un multiplieur-accumulateur, une connexion équivaut à une multiplication et une accumulation. Ainsi, le TMC2208 a une puissance de calcul de 50MOPS (Mega Operations Per Second) mais 25Mcps.

P153385.PDF [Page: 63 of 191]

ces dynamiques par plusieurs ordres de grandeur. Le goulot d'étranglement du système se déplace alors du MSL aux détecteurs, dont le taux de réponse est significativement plus élevé que celui des MSL. Donc, l'utilisation de multiplieurs vecteur-matrice est adéquate pour des cas où les poids d'interconnexion sont déjà trouvés.

Les téléviseurs à cristaux liquides ne sont pas les seuls modulateurs spatiaux de lumière que l'on puisse utiliser dans des architectures à processeurs algébriques optiques. Une autre technologie qui a atteint un certain niveau de maturité en raison de son application dans le domaine militaire est la technologie acousto-optique [18][29][59]. Le tableau 2.2 donne une idée des performances possibles avec la technologie acousto-optique. Ces résultats sont présentés graphiquement à la figure 2.7. On constate que les performances sont supérieures de plusieurs ordres de grandeur aux performances obtenues avec la technologie TVCL. La principale raison est la vitesse de rafraîchissement de la matrice qui, pour la technologie acousto-optique, est de l'ordre du MHz [29], comparativement au kHz (dans le meilleur des cas) pour la technologie TVCL. Ce faible taux de rafraîchissement pour la technologie TVCL est en partie attribuable à la faible vitesse d'alignement des cristaux liquides dans la matrice. Le tableau 2.1 montre que les TVCL à cristaux ferroélectriques ont le potentiel d'être beaucoup plus rapides que les TVCL à cristaux nématiques torsadés. Ces vitesses de rafraîchissement demeurent tout de même loin de celles des matrices acousto-optiques.

Tableau 2.2: Performances de quelques MVM acousto-optiques utilisés dans des applications de réseaux de neurones (adapté de [29]).

Type de processeur	Nombre de neurones	Connexions par seconde (Mcps)[a]	Efficacité (Mcps/w)[a]
Matrice AO 16x16 [29]	16	620/388	171.5/25
Matrice AO 100x100 [59]	100	249000/2490	1100/27.8
Matrice AO 100x100 à ouverture fixe [59]	100	249000	2040
Matrice statique 256x256 [59]	256	6550	208

a. Le premier nombre représente le cas où la matrice est statique alors que le deuxième représente le cas où la matrice est dynamique.

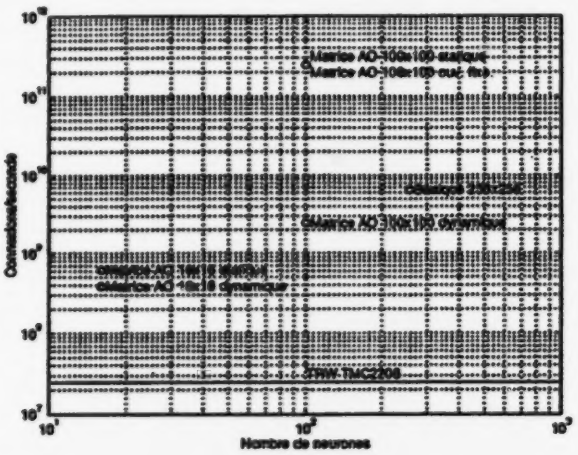


Figure 2.7: Performances pour architectures à processeurs algébriques optiques utilisant des MSL acousto-optiques.

Dans la section suivante, nous allons décrire la deuxième classe d'architectures de réseaux de neurones optiques: les architectures à corrélateurs optiques.

2.3 Architectures à corrélateurs optiques

Une des grandes forces de la technologie optique est sa bande passante. Prenons l'exemple d'une simple lentille convexe. Si on place un objet au foyer de cette lentille et qu'on observe l'image résultante au foyer avant de la lentille, nous retrouvons la transformée de Fourier 2-D de l'objet (voir section 2.1.1 en page 26). Pour rendre compte de la puissance de calcul d'une lentille de Fourier, présentons d'abord un exemple numérique. Supposons que notre objet d'entrée est une image discrète de 512x512 pixels. Un algorithme FFT classique effectue de l'ordre de $N\log_2 N$ calculs (multiplications) [62]. Pour notre image 1024x1024, $N = 2^{18}$ de sorte que $N\log_2 N = 18 \times 2^{18} \approx 4.7 \times 10^6$ calculs. Supposons à présent que nous désirons traiter ces images à cadence vidéo (30 images/seconde). La puissance de calcul du processeur électronique devra donc être d'environ 140 MFLOPs. La carte accélératrice SC-3XL/VME de la compagnie CSPI [17], spécialisée dans le calcul des FFT, permet d'atteindre jusqu'à 200 MFLOPS, mais son prix est sensiblement plus

P153385.PDF [Page: 65 of 191]

élevé que celui d'une lentille. Il apparaît donc clairement que la technologie optique est particulièrement bien adaptée à ce genre de problèmes[1].

Dans la section suivante, nous allons décrire un montage qui met en pratique la propriété de transformation de Fourier des lentilles. Ce montage porte le nom de filtre de Vander Lugt. Par la suite, nous décrirons comment ce montage peut être utilisé pour simuler des architectures neuronales.

2.3.1 Le filtre de Vander Lugt

La figure 2.8 illustre le montage général d'un système de filtrage optique. Ce dernier est constitué de deux lentilles convexes distantes de $2f$ où f est la distance focale des lentilles. La première lentille a pour but de produire au plan de Fourier la transformée de Fourier 2-D du patron lumineux se trouvant sur le plan d'entrée (voir section 2.1.1 en page 26). La deuxième lentille a pour but de transformer à nouveau le patron présent au plan de Fourier vers le plan spatial de sortie. À défaut d'avoir une lentille qui calcule la transformée de Fourier inverse, nous utilisons une deuxième lentille pour calculer à nouveau une TF directe. Le résultat est le même, mis à part une inversion des axes de coordonnées (voir figure 2.8). Si aucun modulateur ne se trouve au plan de Fourier, ce système n'est qu'un simple système d'imagerie dont le but est de reproduire l'image d'entrée (inversée) à la sortie. L'avantage du système devient clair lorsqu'on place un modulateur spatial de lumière au plan de Fourier. Par exemple, supposons que l'on place une diapositive d'un plan noir avec un trou transparent centré sur l'axe optique. Cette opération aura pour effet de laisser passer les basses fréquences de l'image d'entrée tout en coupant les hautes fréquences. Nous retrouverons donc à la sortie une image filtrée passe-bas de l'image d'entrée. Le système agit donc effectivement comme un filtre.

1. Une comparaison plus poussée devrait bien entendu tenir compte du fait qu'une lentille n'effectue pas d'opération à virgule flottante.

P153385.PDF [Page: 66 of 191]

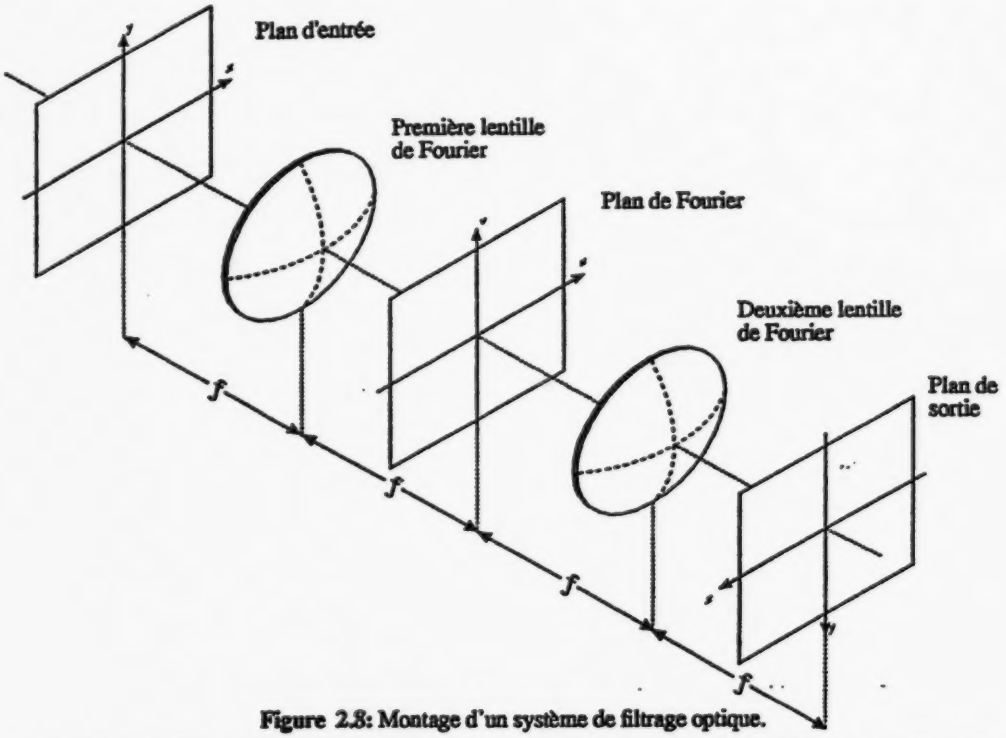


Figure 2.8: Montage d'un système de filtrage optique.

La diapositive étant un simple filtre de transmittance, elle ne peut affecter que l'amplitude du spectre présent au plan de Fourier. Pour moduler à la fois l'amplitude et la phase, nous pouvons placer un hologramme au plan de Fourier. Le montage devient alors un filtre de *Vander Lugt*, en l'honneur de son inventeur [31]. Bien entendu, il faudra replier le système avec le même angle utilisé pour l'écriture de l'hologramme, ceci dans le but de ne considérer que l'ordre qui nous intéresse (voir section 1.2.4 en page 13).

Le théorème de la convolution 2-D nous dit que

$$a(x,y) * b(x,y) = \int_{-\infty}^{\infty}\int a(\xi,\eta)\, b(x-\xi, y-\eta)\, d\xi d\eta \Leftrightarrow A(u,v)\, B(u,v)\,, \qquad (2.9)$$

où * est le symbole de la convolution et $A(u,v)$ et $B(u,v)$ sont les transformées de Fourier 2-D des signaux $a(x,y)$ et $b(x,y)$. Donc, si nous enregistrons sur l'hologramme le

P153385.PDF [Page: 67 of 191]

spectre $B(u, v)$, en plaçant $a(x, y)$ au plan d'entrée, nous obtiendrons la convolution de $a(x, y)$ et $b(x, y)$ au plan de sortie.

Le rôle principal d'un neurone est de calculer la somme pondérée de ses entrants, appelée *entrant net* (voir section 1.1.1). Supposons que $a(x, y)$ représente la matrice des signaux d'entrée d'un neurone et que $b(x, y)$ représente sa matrice d'interconnexion. L'entrant net de ce neurone est alors obtenu en pondérant chaque signal d'entrée a par son poids b correspondant, puis en intégrant le tout. Mathématiquement,

$$\text{net} = \iint a(x, y)\, b(x, y)\, dxdy. \tag{2.10}$$

L'équation précédente est l'expression de la corrélation entre $a(x, y)$ et $b(x, y)$ lorsque ces deux patrons sont centrés. Il est possible d'utiliser le filtre de Vander Lugt de telle sorte que l'image produite au plan de sortie soit la corrélation de $a(x, y)$ et $b(x, y)$. Il suffit d'enregistrer sur l'hologramme non pas $B(u, v)$ mais $B^*(u, v)$. En invoquant le théorème de la convolution, il est possible de démontrer que le patron ainsi produit à la sortie, étant donné une entrée $a(x, y)$, est donné par

$$A(u, v)\, B^*(u, v) \Leftrightarrow \int_{-\infty}^{\infty}\int a(\xi, \eta)\, b(\xi - x, \eta - y)\, d\xi d\eta = a(x, y) \otimes b(x, y), \tag{2.11}$$

où $\otimes$ est le symbole de la corrélation. L'équation (2.10) peut ainsi être obtenue en observant le centre (sur l'axe optique) du patron produit à la sortie, i.e.

$$\text{net} = \int_{-\infty}^{\infty}\int a(x, y)\, b(x, y)\, dxdy = \int_{-\infty}^{\infty}\int a(\xi, \eta)\, b(\xi - x, \eta - y)\, d\xi d\eta \Big|_{x=0,\, y=0}. \tag{2.12}$$

P153385.PDF [Page: 68 of 191]

2.3.2 Considérations d'ordre pratique

Le calcul de corrélations optiques dans le domaine de Fourier profite de la propriété d'invariance en translation de la transformation de Fourier. Ceci implique que, quelle que soit la position de l'image objet dans le plan d'entrée, son spectre sera toujours centré dans le plan fréquentiel. Ce spectre sera donc parfaitement bien aligné avec les autres spectres pré-enregistrés sur l'hologramme. Cependant, toute translation au plan d'entrée se traduira par une translation équivalente au plan de sortie. Ceci pose un sérieux problème lorsqu'il est nécessaire d'échantillonner un point spatial en particulier. Prenons comme exemple l'équation (2.12) qui calcule la corrélation entre deux patrons $a(x, y)$ et $b(x, y)$. Pour que le pic de corrélation soit parfaitement centré sur l'axe optique, il faut que les deux patrons coïncident (i.e que le patron d'entrée soit centré sur l'axe optique du plan d'entrée). Une technique usuelle pour échantillonner le pic de corrélation consiste à placer une plaque opaque munie d'un trou de très faible diamètre centré sur l'axe optique. Il est clair que si l'alignement au plan d'entrée n'est pas parfait, le pic de corrélation (l'entrant net) ne sera pas échantillonné, et donc le calcul de la corrélation sera erroné. Cette contrainte pose donc de sévères limitations sur l'utilisation de ce genre d'architectures. Heureusement, il existe une façon élégante de contourner ce problème. La technique consiste à remplacer la plaquette trouée par un seuilleur opto-électronique ([6], [7]). Ce dernier déterminera lui-même, de façon dynamique, la position du pic de corrélation pour permettre un échantillonnage plus précis.

Une autre limite, imposée cette fois par l'hologramme de Fourier, concerne l'efficacité diffractionnelle. Comme nous l'avons vu dans la section 1.2.4, un hologramme planaire produit, lorsqu'illuminé par un faisceau de référence, 3 ordres de diffraction (un ordre central et deux ordres secondaires). L'énergie diffractée dans l'ordre d'intérêt peut être considérablement plus faible que l'énergie incidente à l'hologramme. De plus, comme

P153385.PDF [Page: 69 of 191]

l'hologramme module l'intensité du faisceau incident, une certaine quantité d'énergie est déjà absorbée par l'hologramme. Une valeur typique d'efficacité diffractionnelle pour un hologramme planaire peut être aussi faible que 6,25% [55]. Cette contrainte peut entraîner des consommations de puissance relativement élevées pour des systèmes comportant des hologrammes planaires.

Il existe deux façons de contourner ce problème. La première est d'utiliser des hologrammes de volume. Les hologrammes de volume ont la particularité de ne produire que deux ordres, ce qui contribue à augmenter l'efficacité diffractionnelle. Mais la méthode la plus efficace est sans aucun doute l'utilisation d'hologrammes dits «de phase» [43]. Un hologramme de phase module uniquement la phase d'un faisceau incident, sans en affecter l'amplitude. Leur efficacité en transmittance est donc près de 100% puisque l'hologramme ne produit, en principe, aucune atténuation. Ce genre d'approche a cependant une utilité limitée. Par exemple, dans un corrélateur de Vander Lugt utilisant un hologramme de phase, seule la phase[1] $\varphi(u, v)$ de la fonction de transfert $H(u, v) = |H(u, v)|e^{j\varphi(u, v)}$ est enregistrée, de sorte que l'intensité du pic de corrélation (en tenant compte de l'accroissement d'efficacité) ne sera pas la même qu'avec un corrélateur utilisant $H(u, v)$ comme filtre. Donc, si le but est de calculer une corrélation exacte, l'approche initiale est de rigueur, tandis que si le but est simplement de détecter une forte ressemblance entre deux patrons, l'approche utilisant les hologrammes de phase est plus appropriée.

Finalement, un important atout en faveur de ce genre d'architectures est qu'elles sont parmi les seules qui peuvent être réalisées entièrement en optique (i.e. sans aucune conversion optique/électronique). En effet, bien souvent, le chemin optique du système est replié sur lui-même de façon a introduire une rétroaction. Le système agit alors comme

1. Il est également possible d'effectuer une modulation en phase avec $|H(u, v)|$.

P153385.PDF [Page: 70 of 191]

une mémoire associative [44], [64], [68], [63]. Le patron d'entrée initial est introduit dans le système par l'entremise de la valve optique qui agit ici comme un transducteur. Bien entendu, si le patron d'entrée n'est pas disponible optiquement (i.e. sous forme d'une image), il devra être convertit en signal optique avant d'être introduit dans le système. Nous décrirons plus en détail les mémoires associatives dans la section suivante.

2.3.3 Performances

Pour calculer les performances des architectures à corrélateurs optiques, nous allons nous baser sur la mémoire associative de Psaltis *et al.*[44][64][68]. Historiquement, leur prototype était le premier à être réalisé exclusivement avec des composants optiques. Leur architecture repose sur une version 2-D du populaire modèle de Hopfield [41] que nous allons maintenant décrire brièvement.

Le modèle de Hopfield 2-D est un agencement planaire de $N \times N$ neurones complètement interconnectés. Ce réseau agit à titre de mémoire auto-associative, c'est-à-dire que lorsqu'on lui présente une version bruitée d'un patron qu'il a mémorisé, son état convergera vers le patron mémorisé. Contrairement aux autres réseaux qui nécessitent une phase d'apprentissage pour ajuster leurs poids d'interconnexion, la matrice de poids dans le modèle de Hopfield doit être déterminée *a priori* à partir des patrons que l'on désire faire apprendre au réseau. Si nous dénotons par $f_m(x, y)$ le $m^{\text{ème}}$ patron bidimensionnel d'un ensemble de M patrons au total, les poids d'interconnexion ω sont déterminés par la *loi du produit externe* ou *loi de Hebb* de la façon suivante

$$\omega(x, y; \xi, \eta) = \sum_{m=1}^{M} f_m(x, y) f_m(\xi, \eta) . \qquad (2.13)$$

Remarquez que la matrice ω est un tenseur 4-D (i.e. matrice 2-D de matrices 2-D).

P153385.PDF [Page: 71 of 191]

Les neurones dans le modèle d'Hopfield effectuent un seuillage non-linéaire sur leur entrant net. Chaque neurone dans la matrice 2-D recoit des entrants des N^2 neurones, pondérés par N^2 poids. Dénotons l'état des N^2 neurones au temps t par $f(\xi, \eta, t)$. L'entrant net d'un neurone à la position (x, y) est donné par

$$\text{net}_{(x,y)} = \iint \omega(x, y; \xi, \eta) f(\xi, \eta, t)\, d\xi d\eta \,. \tag{2.14}$$

Ainsi, l'équation d'activation du réseau (la sortie des neurones, après le seuillage) est

$$\hat{f}(x, y, t) = g\left\{\iint \omega(x, y; \xi, \eta) f(\xi, \eta, t)\, d\xi d\eta\right\}, \tag{2.15}$$

où g est une fonction de seuillage non-linéaire. En remplaçant $\omega(x, y; \xi, \eta)$ dans (2.15) par l'expression (2.13) et en réorganisant l'ordre d'intégration, (2.15) devient (voir développement dans [44]).

$$\hat{f}(x, y, t) = g\left\{\sum_{m=1}^{M} f_m(x, y)\left[\iint f_m(\xi, \eta) f(\xi - x, \eta - y, t)\, d\xi d\eta\right]\Big|_{x=0, y=0}\right\}. \tag{2.16}$$

L'expression entre crochets représente l'intercorrélation entre le patron à l'entrée et le patron $f_m(\xi, \eta)$. En l'évaluant à $x = 0, y = 0$, on obtient un scalaire égal au produit interne entre $f_m(\xi, \eta)$ et $f(\xi, \eta)$ (voir (2.12)). Ce scalaire mesure la similarité entre $f_m(\xi, \eta)$ et $f(\xi, \eta)$. L'expression entre accolades peut alors être vue comme une moyenne pondérée de tous les patrons stockés, les facteurs de pondération étant les similarités (i.e. produits internes) entre $f(\xi, \eta)$ et tous les $f_m(\xi, \eta)$. Par exemple, supposons que le patron d'entrée $f(\xi, \eta) = f_k(\xi, \eta)$, le $k^{\text{ème}}$ patron stocké en mémoire. Le facteur de pondération pour ce patron sera égal à l'autocorrélation de $f_k(\xi, \eta)$ évalué à l'origine. D'après la définition de l'autocorrélation, ce facteur sera le plus élevé des M facteurs. Par conséquent la moyenne sera dominée par $f_k(\xi, \eta)$. Autrement dit, l'image au plan de sor-

P153385.PDF [Page: 72 of 191]

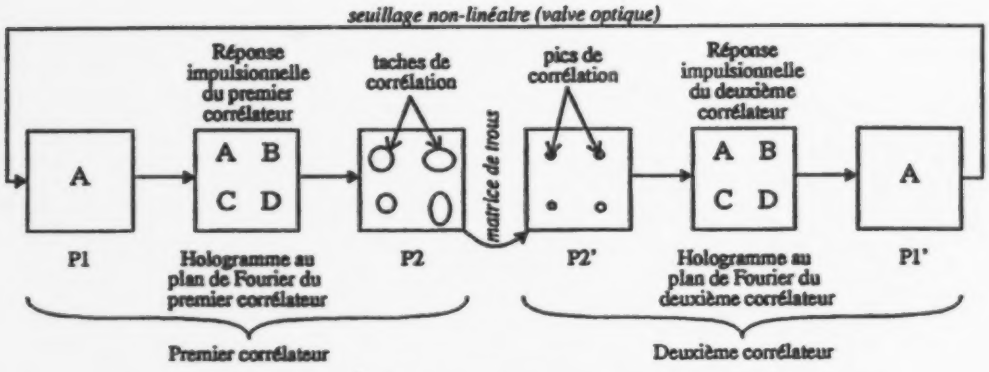


Figure 2.9: Principe d'opération de la mémoire associative de Psaltis (d'après [44]).

tie ressemblera beaucoup à $f_k(\xi, \eta)$, mais elle sera superposée à des versions atténuées des $M-1$ autres patrons. En effectuant un seuillage non-linéaire, la contribution des $M-1$ autres patrons sera davantage atténuée. Comme la sortie est présentée à l'entrée par rétroaction, après plusieurs itérations, l'image de sortie convergera vers $f_k(x, y)$.

Le principe de fonctionnement de l'architecture de Psaltis est illustré à la figure 2.9. Elle est constituée de deux corrélateurs de Vander Lugt montés en boucle pour produire l'équation (2.16). Le premier corrélateur effectue l'opération entre crochets (voir développement à la section 2.3.1). Le deuxième corrélateur effectue la moyenne pondérée. L'hologramme au plan de Fourier des deux corrélateurs renferme l'information spectrale sur les M patrons[1]. La réponse impulsionnelle des corrélateurs est illustrée à la figure. L'image d'entrée est d'abord présentée au plan P1 (le plan d'entrée). Le premier corrélateur calcule ensuite les M corrélations de l'image d'entrée avec les images mémorisés dans l'hologramme, et les affiche au plan P2. Les facteurs de pondération sont ensuite échantillonnés au plan P2' en plaçant une plaque dans laquelle M trous ont été usinés. Chacun de ces trous est vis-à-vis l'axe optique (i.e. $x = 0, y = 0$) de chaque tache de corrélation (une

1. L'architecture permet de stocker seulement quatre patrons.

P153385.PDF [Page: 73 of 191]

tache pour chaque patron) de façon à ne laisser passer que l'information sur le produit interne. Ces quatres points lumineux agissent alors comme quatre fonctions de Dirac au plan d'entrée du deuxième corrélateur (P2'). Chacune reconstruira au plan P1' l'image correspondant à sa position avec une intensité proportionnelle au facteur de pondération. Ainsi, au plan de sortie du deuxième corrélateur (P1'), on retrouvera la moyenne pondérée des M images. Dans l'exemple de la figure 2.9, comme nous avons présenté la lettre "A" à l'entrée, la corrélation avec le "A" mémorisé est forte. Conséquemment, la Dirac la plus intense au plan P2' sera centré sur la position du "A" en mémoire (représenté par un point noir sur la figure). Cette Dirac reconstituera donc le "A" au plan P1'. Le plan de neurones (i.e. les fonctions de seuillage) est simulé par une valve optique placée au plan de sortie du deuxième corrélateur P1' (et incidemment au plan d'entrée du premier corrélateur, P1). Pour une description plus détaillée, veuillez consulter les références [44] et [68].

L'architecture réalisée effectue considérablement plus d'opérations que ce que le modèle de Hopfield requiert. Pour s'en convaincre, il suffit de comparer la complexité des équations (2.15) et (2.16). Cette dernière effectue M corrélations 2-D pour ensuite n'échantilloner que la valeur centrale. Pour évaluer la performance *effective* de l'architecture en tant que réseau de neurone, il est primordial de ne considérer *que* les opérations pertinentes au réseau (i.e équation (2.15)). Toutes les opérations supplémentaires effectuées dans l'équation (2.16) l'auront été dans le but de permettre une simplification de la réalisation optique, mais elles n'apportent rien de plus au résultat final.

Selon l'équation (2.15), pour chaque neurone (ou pixel) à la position (x, y), une intégrale double doit être effectuée (N^2 opérations). Donc, pour une matrice de $N \times N$ neurones, un total de N^4 connexions devront être calculées. Le nombre de connexions par seconde dépendra de la vitesse avec laquelle les N^4 connexions sont réalisées.

P153385.PDF [Page: 74 of 191]

Les performances du système sont limitées uniquement par la valve optique qui agit comme un plan de neurones. Le tableau 2.3 suivant énumère les caractéristiques de quelques valves optiques, ainsi que leurs métriques pour une application de réseau de neurones. La figure 2.10 illustre graphiquement les données du tableau 2.3.

Tableau 2.3: Caractéristiques de quelques valves optiques pour la mise en œuvre d'architectures à corrélateurs optiques.

Type de valve optique	Type de cristaux liquides	Résolution (à 50% de la MTF)	Taux maximum de rafraîchissement	Zone active	Nombre de pixels (ou neurones)	Connexions par seconde (cps)
Hughes H-4160 [45]	CLNT[a]	16 pl/mm	5 Hz	25x25 mm^2	160 000	$128x10^9$
Control Optics[b] [15]	CLNT	20 pl/mm	14 Hz	~962 mm^2[c]	385 000	$2.075x10^{12}$
NTT [26], [27]	CLF[d]	~100 pl/mm	~1000 Hz[e]	12x12 mm^2	1 440 000	$2.07x10^{15}$
Seiko LAPS [76]	CLF	~100 pl/mm[f]	1000 Hz	18x18 mm^2	3 240 000	$10.5x10^{15}$

a. CLNT: Cristaux Liquides Nématiques Torsadés.

b. Évalué sommairement par l'Institut National d'Optique [21].

c. Diamètre de 35 mm^2.

d. CLF: Cristaux Liquides Ferroélectriques.

e. Meilleurs résultats à de tels taux de rafraîchissement. Cependant, il n'est pas évident que la résolution et le contraste restent aussi bons.

f. Résolution en mode niveau de gris [76].

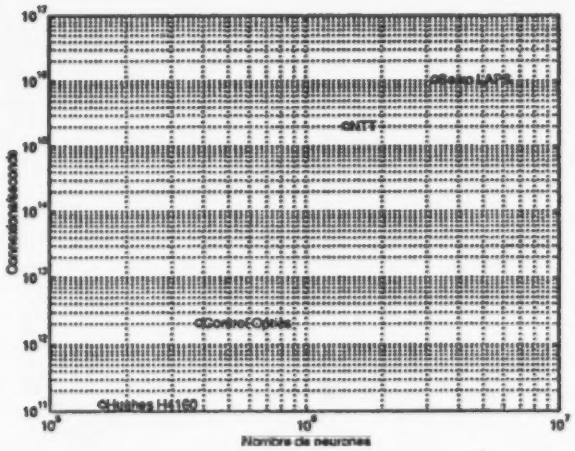


Figure 2.10: Métriques de performance pour architectures à corrélateurs optiques utilisant des valves optiques.

P153385.PDF [Page: 75 of 191]

L'architecture de Psaltis est un excellent exemple d'une utilisation optimale des avantages d'une technologie pour la mise en œuvre d'un réseau donné. En effet, le réseau de Hopfield est tel qu'il est possible de décomposer sa dynamique en plusieurs opérations élémentaires qui se réalisent merveilleusement bien en optique (équations (2.15) à (2.16)). La source de ces avantages provient de la symétrie de la matrice de poids du réseau et de l'utilisation de la loi du produit externe pour l'apprentissage (loi de Hebb). En effet, si ces conditions n'avaient pas été respectées, la décomposition (2.16) n'aurait pas été possible et une architecture sans doute beaucoup plus complexe aurait résulté. Il est donc important de résister à la tentation de se laisser éblouir par les performances du graphique 2.10. Ces chiffres ne disent pas, par exemple, que le modèle est incapable d'apprentissage continu (i.e. les poids doivent être déterminés *a priori*), ni qu'il souffre de la faible capacité de mémorisation inhérente au modèle de Hopfield. La conclusion à tirer est que pour la mise en œuvre de mémoires associatives inspirées du modèle de Hebb, les architectures à corrélateurs optiques offrent les meilleures performances. Aucune autre architecture ne peut espérer rivaliser avec cette dernière pour ce genre de problèmes.

Une tendance grandissante chez les ingénieurs d'architectures neuronales est de s'inspirer initialement de modèles neuronaux pour réaliser un prototype, mais de modifier par après l'architecture pour contourner certains problèmes inhérents au modèle sous-jacent. Bien souvent, l'architecture résultante ne ressemble plus au modèle initial mais est beaucoup plus utile du point de vue pratique. Le fait qu'une architecture matérielle ressemble peu au modèle neuronal duquel elle est inspirée ne pose pas un problème en soi. Si l'architecture résultante est plus performante (par son utilisation judicieuse des bienfaits d'une technologie par exemple) et qu'elle permet de résoudre de vrais problèmes pratiques, son existence est pleinement justifiée. Un exemple est le fameux réseau «*backpropagation*». Beaucoup de chercheurs ont critiqué ce réseau sous prétexte qu'il imitait mal le fonction-

P153385.PDF [Page: 76 of 191]

nement réel des neurones biologiques. Cependant ce réseau a fait ses preuves pour des applications de classification et d'association («*mapping*»).

Une contrainte non négligeable est le coût. Les valves optiques sont parmi les MSL les plus dispendieux. Leur prix varient de 3 000$ à 16 000$ U.S. en 1993 [15],[57].

La section suivante décrit le troisième et dernier type d'architectures neuronales optiques: les architectures à interconnexions holographiques.

2.4 Architectures à interconnexions holographiques

2.4.1 Principe

Le principe de l'interconnexion holographique est relativement simple. Supposons que nous désirons interconnecter N neurones avec N autres neurones. Disposons ces neurones sur deux plans parallèles. Le principe consiste à placer entre ces deux plans un matériau dont les propriétés optiques font en sorte que la communication entre les plans soit spécifiée. Les connexions sont réalisées en faisant dévier avec pondération les faisceaux émanant du plan d'entrée vers les neurones appropriés du plan de sortie. Cette approche est illustrée à la figure 2.11.

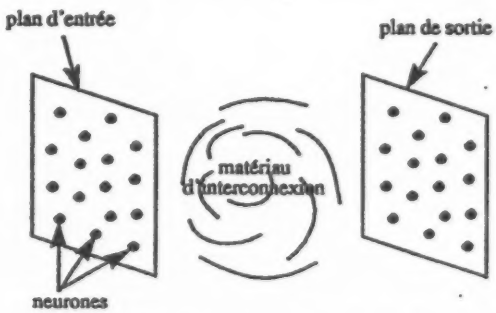


Figure 2.11: Principe d'interconnexion holographique.

P153385.PDF [Page: 77 of 191]

Ainsi, tout neurone du plan d'entrée aura une connexion avec chaque neurone du plan de sortie. Avec N neurones à l'entrée et N neurones à la sortie, le nombre total d'interconnexions requises est N^2.

Un matériau bien connu possédant les propriétés mentionnées plus haut est l'hologramme (voir section 1.2.4 en page 13). Dans la section suivante, nous verrons quelles contraintes nous devons imposer à ces hologrammes pour spécifier les interconnexions.

2.4.2 Topologies d'interconnexion et degrés de liberté

Examinons à présent de plus près le problème d'interconnexion dans le but de mieux comprendre les limites physiques de cette approche. Définissons tout d'abord le concept de degré de liberté à l'aide d'un exemple. Lorsque nous écrivons sur une feuille de papier, notre crayon ne peut se déplacer que selon deux directions principales: verticale ou horizontale. On dira alors que la feuille permet deux degrés de liberté. Cependant, lorsque nous écrivons dans un livre, nous pouvons profiter d'un troisième degré de liberté: la profondeur (i.e. les différentes pages du livre). Nous avons donc alors accès à un total de trois degrés de liberté. En d'autres termes, le nombre de degrés de liberté correspond à la *dimensionnalité* de l'espace géométrique. Dans le monde physique, nous sommes limités à trois dimensions.

Lorsque nous utilisons un matériau physique pour stocker de l'information (dans notre cas, des interconnexions), la quantité maximale d'information que l'on puisse espérer stocker est fonction des degrés de liberté du matériau. Prenons comme exemple un hologramme planaire (deux degrés de liberté). Définissons N comme étant le nombre de points résolvables[1] selon une dimension. Cet hologramme aura donc N^2 points résolvables. Si

1. Un point résolvable est la plus petite dimension physique dans laquelle il est possible de stocker de l'information. Dans un hologramme volumique de volume V, le nombre de points résolvables est V/λ^3 où λ est la longueur d'onde utilisée.

P153385.PDF [Page: 78 of 191]

cet hologramme est utilisé pour stocker des interconnexions, on peut espérer stocker de l'ordre de N^2 interconnexions. De la même façon, si on utilise un hologramme volumique, nous aurions à notre disposition $N \times N \times N = N^3$ points résolvables pour une capacité de stockage de l'ordre de N^3. Cette capacité de stockage représente une limite physique puisque, rappelons-le, nous ne disposons que de trois dimensions dans le monde réel.

Revenons à présent à notre problème d'interconnexion. Considérons différentes topologies d'interconnexion et déterminons la dimensionnalité requise pour réaliser ces topologies:

- n neurones d'entrée reliés avec n neurones de sortie: n^2 interconnexions (2 degrés);
- n neurones d'entrée reliés avec n^2 neurones de sortie: n^3 interconnexions (3 degrés);
- n^2 neurones d'entrée reliés avec n^2 neurones de sortie: n^4 interconnexions (4 degrés).

Les deux premières topologies sont physiquement réalisables puisque la dimensionnalité requise est inférieure ou égale à 3. La troisième, par contre, n'est pas directement réalisable physiquement puisqu'elle requiert quatre dimensions. Pourtant, le problème d'interconnecter un plan (n^2) de neurones avec un autre plan (n^2) semble relativement simple. Il existe, bien entendu, des compromis qu'il est possible de faire pour obtenir cette capacité d'interconnexion. Comme on peut s'y attendre, ce compromis entraînera un sacrifice d'espace (i.e. les dimensions du réseau résultant seront plus grandes).

Selon le choix du compromis, il existe deux approches principales au problème d'interconnexion n^4. La première approche, prônée par Psaltis, consiste à réduire la dimensionnalité à l'entrée et à la sortie, de sorte que la dimensionnalité requise pour l'interconnexion de l'entrée à la sortie n'excède pas 3. Il s'agit tout d'abord de s'assurer que les n^4 interconnexions puissent être contenues dans un volume avec N^3 points résolvables. Pour ne pas excéder la dimensionnalité du volume, on fait en sorte de n'utiliser que 3/2 degrés de liberté à l'entrée et à la sortie, de sorte que $N^{3/2} \times N^{3/2} = N^3$. Notez

P153385.PDF [Page: 79 of 191]

que si on tente d'enregistrer N^4 connexions dans le volume (i.e. N^2 neurones au plan d'entrée et de sortie), il y aura inévitablement de l'interférence entre les connexions («cross-talk»). Cependant, lorsqu'on se contente de $N^{3/2}$ neurones au plan d'entrée et de sortie, Psaltis a démontré [68] qu'il est possible de disposer ces neurones de telle sorte à ce que l'interférence entre les connexions soit nulle. Ces agencements portent le nom de *grilles d'échantillonnage fractales* [68].

La deuxième approche populaire, prônée par Caulfield, n'impose pas de modifications au plan d'entrée et de sortie, mais elle les impose au médium d'interconnexion. Supposons qu'une matrice carrée de n^2 neurones doive communiquer avec une autres matrice carrée de n^2 neurones. Un hologramme à deux dimensions peut spécifier les interconnexions entre un neurone du plan d'entrée et tous les n^2 neurones du plan de sortie car $n^0 \times n^2 = n^2$, donc deux degrés de liberté. Donc, pour spécifier les connexions avec tous les n^2 neurones du plan d'entrée, il suffit tout simplement de créer un matrice d'hologrammes 2-D où chaque hologramme spécifiera n^2 connexions. Nous aurons alors

$$n^2 \text{holograms} \times n^2 \frac{\text{interconnexions}}{\text{hologram}} = n^4 \text{interconnexions} . \qquad (2.17)$$

Cette approche porte le nom de multiplexage spatial, car elle divise l'espace en plusieurs régions indépendantes, chacune accueillant un hologramme planaire différent.

Les deux prochaines sections décriront plus en détail ces deux approches.

2.4.3 Multiplexage spatial

La figure 2.12 illustre le montage de Caulfield, tel que décrit dans [13]. Le principe est relativement simple. Une matrice de $N \times N$ hologrammes est éclairée sur toute sa surface par un même faisceau de référence. $N \times N$ images des matrices de poids seront alors imagées sur le MSL réflectif (extrême droite sur la figure) qui affiche le patron d'entrée 2-D.

P153385.PDF [Page: 80 of 191]

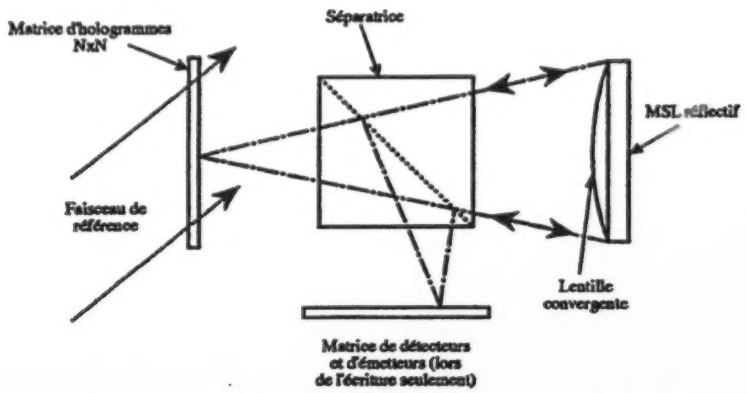


Figure 2.12: Montage optique du système d'interconnexion de Caulfield.

Le résultat du produit entre les matrices et le patron d'entrée sont par la suite concentrés en différents points sur le plan de sortie par la lentille convergente. Donc, à chaque point (ou neurone) du plan de sortie correspond un sous-hologramme qui spécifie ses interconnexions.

La spécification des interconnexions se fait optiquement, en renversant la direction de propagation des faisceaux lumineux. Cette programmation se fait par exposition multiple de l'hologramme. Pour chaque exposition, les N^2 connexions d'un neurone de sortie donné sont spécifiées en parallèle. Une source lumineuse à la position de ce neurone dans la plan de sortie éclaire le MSL sur lequel sa matrice de poids a préalablement été écrite. Le faisceau modulé est ensuite réfléchi par le MSL jusqu'à l'hologramme. Une version conjuguée du faisceau de référence illumine alors l'hologramme et le patron d'interférence résultant est alors enregistré.

2.4.4 Multiplexage volumique

La figure 2.13 illustre un montage possible pour une architecture à interconnexions holographiques faisant appel au multiplexage volumique [68]. Il est constitué de trois plan: le plan d'entrée, le plan de sortie et le plan d'apprentissage. Le plan d'apprentissage

P153385.PDF [Page: 81 of 191]

est symétriquement opposé au plan de sortie. Sa fonction est de spécifier les interconnexions entre le plan d'entrée et le plan de sortie. L'interconnexion entre deux points se fait comme suit: le neurone d'entrée émet un faisceau lumineux qui est transformé en onde plane par la lentille de Fourier. L'image du neurone de sortie au plan d'apprentissage émet également un faisceau qui lui aussi est converti en onde plane. Ces deux ondes interfèrent dans le cristal pour former un grillage sinusoïdal. Si le cristal est exposé suffisament longtemps, le patron d'interférence sera ainsi enregistré. Lorsqu'un faisceau de lumière émanera du même neurone d'entrée (le plan d'apprentissage étant inactif), ce dernier sera dévié puis convergera vers le neurone de sortie pour lequel la connexion a été spécifiée.

Notez qu'il n'est pas nécessaire de spécifier les connexions une à une. Pour chaque neurone d'entrée, les connexions avec tous les neurones de sortie peuvent être spécifiées d'un seul coup et enregistrées avec une seule exposition. Le nombre total d'expositions nécessaires pour spécifier toutes les connexions est donc égal au nombre de neurones d'entrée.

2.4.5 Considérations d'ordre pratique

L'architecture de Caulfield souffre d'au moins deux problèmes. Le premier est que pour atteindre un nombre suffisant d'interconnexions, les dimensions de la matrice d'hologrammes peuvent devenir excessivement grandes. Par exemple, pour réussir à spécifier 10^{12} interconnexions avec une matrice 1000×1000 d'hologrammes de 1000×1000 pixels, en supposant une résolution de 500 pixels/mm, la matrice d'hologrammes devrait avoir une surface de $2\text{m} \times 2\text{m}$!. Un hologramme d'une telle dimension est pratiquement irréalisable et serait très difficile à aligner. Une estimation plus conservatrice permettrait de réaliser 2.5×10^8 interconnexions avec une matrice de $12.8 \times 12.8\text{cm}$ constituée de 128×128 sous-hologrammes.

P153385.PDF [Page: 82 of 191]

Le deuxième problème est que les poids des interconnexions doivent être fixes puisque les hologrammes eux-mêmes sont fixes. L'utilisation d'une TVCL au lieu de l'hologramme n'est pas une option, car cela demandrait un effort de calcul trop grand pour la synthèse par ordinateur et l'affichage des hologrammes d'interconnexion.

Le principal atout de cette architecture est sa simplicité et son utilisation de composantes relativement peu dispendieuses (hologramme planaire, séparatrice, matrice de détecteurs, MSL réflectif et une lentille non-critique). De plus, l'auteur suggère qu'une fois exposé et développé, l'hologramme pourrait aisément être cloné pour la production en masse. Mais si les poids doivent être adaptatifs, les cristaux photoréfractifs présentent un avantage évident.

Le principal avantage en faveur de l'utilisation de cristaux photoréfractifs, éléments constituants de l'architecture de Psaltis, est qu'ils permettent la mise en œuvre d'algorithmes d'apprentissage en temps réel. Cet apprentissage s'exécute en ré-exposant le cristal à

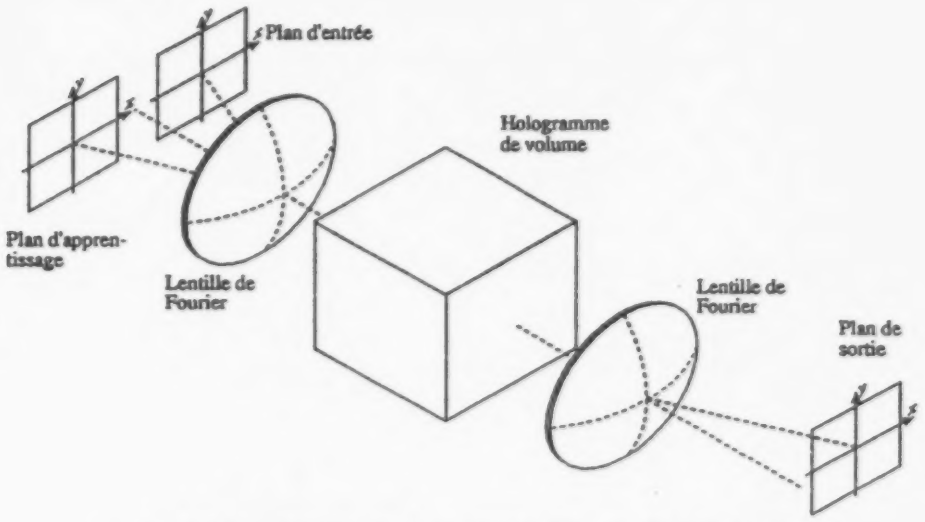


Figure 2.13: Montage optique du système d'interconnexion de Psaltis [68].

P153385.PDF [Page: 83 of 191]

chaque itération. Un autre avantage est que, contrairement aux architectures utilisant des MSL adressés optiquement comme les TVCL, l'écriture du cristal se fait optiquement, en faisant interférer deux faisceaux cohérents. Donc, la mise à jour peut se faire en parallèle, ce qui devient important lorsque le nombre d'interconnexions croît.

Un problème inhérent à la technique d'enregistrement du cristal est que la force d'une interconnexion ne peut être modifiée sans modifier la force des autres interconnexions déjà enregistrées sur le cristal. L'expression décrivant l'amplitude d'un hologramme, c'est-à-dire la différence entre la charge la plus élevée et la plus faible dans l'hologramme, est donnée par l'équation suivante: [68]

$$A_m = A_0\left[1-\exp\left(\frac{-t_m}{\tau_r}\right)\right]\exp\left(-\sum_{m'=m+1}^{M}\frac{t_{m'}}{\tau_e}\right), \tag{2.18}$$

où A_m et t_m sont l'amplitude et le temps d'exposition de la $m^{\text{ème}}$ exposition respectivement ($m = 1, \ldots, M$), τ_r et τ_e sont les constantes de temps d'exposition pour l'enregistrement et l'effacement respectivement et A_0 est l'amplitude de saturation de l'hologramme. Le terme entre crochet contribue à augmenter l'amplitude de l'hologramme, alors que l'exponentielle de droite, qui est fonction du temps d'exposition après l'enregistrement, contribue à la diminuer. Une situation désirable est que les amplitudes des M hologrammes soient égales dans le but de préserver les proportions. Il est possible de démontrer que si le temps d'exposition du $m^{\text{ème}}$ hologramme est $t_m = \tau_e/m$, alors l'amplitude A_m (quelle que soit la valeur de m) peut être approximée par (voir [67])

$$A_m \approx \frac{\tau_e}{\tau_r}\frac{A_0}{M} \quad \Rightarrow \quad M \approx \frac{\tau_e}{\tau_r}\frac{A_0}{A_m}. \tag{2.19}$$

L'équation (2.19) nous permet de tirer une conclusion importante. Le nombre maximum d'hologrammes qu'il soit possible d'enregistrer est déterminé par le rapport des

P153385.PDF [Page: 84 of 191]

constantes de temps d'écriture et de lecture et par le rapport entre l'amplitude de chaque hologramme et l'amplitude de saturation. Plus A_m est petit comparativement à A_0, plus on sera en mesure d'emmagasiner d'hologrammes dans le cristal. Comme l'efficacité diffractionelle est proportionnelle au carré de l'amplitude des hologrammes, leur efficacité diffractionelle sera inversement proportionnelle au carré du nombre M d'expositions. Ceci a pour effet de limiter le nombre d'hologrammes qu'il est possible de stocker dans le cristal, car un hologramme de trop faible intensité risque soit d'être perturbé par du bruit optique, soit de diffracter une énergie tellement faible qu'un détecteur sera incapable de la détecter correctement.

2.4.6 Performances

Très peu de métriques de performances sont disponibles pour les architectures utilisant des cristaux photoréfractifs. La plupart du temps, les auteurs font mention du nombre impressionnant d'interconnexions qu'il est possible d'obtenir (entre 10^8 et 10^{10} par cm^3 [67]) et de la possibilité d'incorporer des algorithmes d'apprentissage. L'avantage de la densité d'interconnexion est inattaquable. Cependant l'avantage de pouvoir incorporer des algorithmes d'apprentissage est loin d'être démontré. La dynamique du procédé d'enregistrement du cristal (équation (2.18)) entraîne une foule de problèmes dont les plus importants sont décrits dans ce qui suit.

Le premier problème concerne la relation quadratique entre l'efficacité diffractionnelle et le nombre d'expositions que nous avons décrite dans la section précédente. Ceci a pour effet de limiter la densité d'interconnexion en imposant une limite pratique sur le nombre d'expositions permises. Une technique astucieuse permet cependant de changer la relation quadratique en relation linéaire. La technique est de ré-enregistrer périodiquement les hologrammes déjà enregistrés, mais cette fois en exposant le cristal suffisament longtemps pour que sa saturation soit atteinte. Ainsi, là où il était possible d'enregistrer 100 holo-

P153385.PDF [Page: 85 of 191]

grammes avec la technique précédente, il est maintenant possible d'en enregistrer 10 000. Cette technique requiert toutefois un temps d'exposition beaucoup plus long.

Le second problème est que le nombre d'expositions doit être connu *a priori* pour pouvoir calculer la série des temps d'exposition (i.e. l'ensemble des temps d'expostition $\{t_m\}$ pour tous les hologrammes) optimale. La cédule optimale est celle qui enregistrera tous les hologrammes avec la même amplitude. Comme nous l'avons mentionné dans la section précédente, pour rencontrer cette condition, il faut que le temps d'exposition du $m^{\text{ème}}$ hologramme soit $t_m = \tau_e / m$. Mok *et al.* [58] ont calculé expérimentalement une constante de temps d'écriture $\tau_r = 1000$ secondes (pour une illumination de 100mW/cm^2) pour un cristal de $LiNbO_3$ et un rapport $\tau_r / \tau_e = 1.05$, ce qui implique une constante d'effacement $\tau_e \approx 1000$ secondes. Dans leur expérience, Mok *et al.* ont enregistré 500 hologrammes dans leur cristal. Le temps total d'exposition, en supposant une schédule optimale est donné par

$$t_T = \sum_{m=1}^{M} \frac{\tau_e}{m} = 1000 \sum_{m=1}^{500} \frac{1}{m} = 1000 \times 4{,}5 = 4500\text{s} \; . \qquad (2.20)$$

Donc, le temps total d'exposition est de 75 minutes! Bien entendu, il est possible d'utiliser des temps d'exposition plus faibles, mais ce sera au détriment d'une efficacité diffractionnelle réduite. Une autre solution serait d'augmenter l'illumination lors de l'enregistrement pour diminuer la constante de temps d'écriture (et, par le fait même, d'effacement).

Un autre problème est que la lecture d'un hologramme déjà enregistré contribue à effacer partiellement tous les autres hologrammes (y compris celui qui est lu). Cependant, comme la constante de temps d'effacement est généralement très élevée, si la lecture est d'une durée suffisamment faible, la dégradation sera relativement faible.

P153385.PDF [Page: 86 of 191]

Le coût d'un cristal de $LiNbO_3$ de $1cm^3$ se situe aux alentours de 2200$ U.S. [21]. Bien que la densité de stockage de cet hologramme soit grande, le prix est tout de même relativement élevé si on le compare avec le prix d'hologrammes planaires. Un hologramme planaire généré par ordinateur de résolution 50pl/mm coûte aussi peu que 1$ pour une surface de 8.5x11 pouces carrés! Bien entendu, les dimensions de l'hologramme planaire seront plus grande que celle du cristal pour une valeur du produit espace-bande constante.

La solution qui sans doute offrira les meilleures performances est de réconcilier la dynamique de l'écriture des cristaux avec celle des algorithmes d'apprentissage [67]. En effet, les algorithmes d'apprentissage proposés par les neurophysiologues et les mathématiciens ne l'ont certes pas été avec le souci de leur simplicité de réalisation à l'aide des technologies optiques. C'est en développant de nouveaux algorithmes qui se marieront bien avec la dynamique d'écriture inhérente aux cristaux photoréfractifs que ce genre d'architectures pourra espérer un jour rivaliser avec les architectures électroniques (numériques).

Les trois dernières sections ont décrit les principales architectures de réseaux de neurones optiques, soit les architectures à processeurs algébriques optiques, les architectures à corrélateurs optiques ainsi que les architectures à interconnexions holographiques. Pour ces trois architectures, nous avons discuté des considérations d'ordre pratique ayant trait à leur mise en œuvre ainsi que des performances.

Le tableau comparatif 2.4 a pour but de résumer les principaux avantages et inconvénients de chaque type d'architecture selon les points de vue pratique et performance. Dans le chapitre suivant, nous allons nous intéresser aux réalisations électroniques de réseaux de neurones.

P153385.PDF [Page: 87 of 191]

Tableau 2.4: Résumé des avantages et inconvénients des trois types d'architectures optiques selon les points de vue pratique et performance.

Architecture		Point de vue	Avantages	Inconvénients
À processeurs algébriques optiques		Pratique	• Architectures versatiles • Possibililité d'augmenter la précision des calculs • Possibilité d'effectuer des calculs avec des nombres négatifs • Possibilité d'incorporer l'apprentissage simultané • Permet d'émuler un grand nombre de types de réseaux • Coût relativement faible	• Problèmes de synchronisation • Problèmes d'alignement dus aux aberrations sphériques des lentilles • Problèmes de vignetage • Ne traite que des patrons unidimensionnels
		Performance	• Vitesse accrue lorsque la matrice des poids est statique • Vitesse accrue avec matrices acousto-optiques (statiques ou dynamiques)	• Vitesse limitée, surtout en mode de précision accrue • Processeurs entièrement électroniques souvent plus performants • Simule un faible nombre de neurones • Topologies d'interconnexion limitées
À corrélateurs optiques		Pratique	• Architectures parmi les seules qui peuvent être réalisées entièrement avec des composantes optiques • Peut aisément traiter des patrons d'entrée bidimensionnels	• Problèmes d'échantillonnage des pics de corrélation • Faible efficacité diffractionnelle pour les hologrammes, donc consommation de puissance élevée • Possibilité d'émuler qu'un nombre très restreint d'architectures neuronales • Coût peut être élevé si des composants dispendieux sont utilisés (cristal photoréfractif, valve optique)
		Perf.	• Vitesse très élevée • Permet de simuler beaucoup de neurones	
À interconnexions holographiques	multiplexage spatial	Pratique	• Architectures très simples • Mise à jour des poids effectuée optiquement (et en parallèle) • Coût relativement faible	• Taille de l'hologramme peut devenir démesurément grande • Problèmes d'alignements lorsque l'hologramme est trop grand • Problème de codage des hologrammes
		Perf.	• Possibilité de vitesse élevée	
	multiplexage volumique	Pratique	• Profite du stockage d'information en trois dimensions • Possibilité d'apprentissage «on-line» • Mise à jour des poids effectuée optiquement (et en parallèle) • Permet une grande flexibilité quant à la topologie d'interconnexion	• Impossibilité de modifier une interconnexion sans modifier les autres • Lecture du cristal dégrade l'hologramme • Coût élevé • Problèmes d'interférence («crosstalk») • Fragilité mécanique • Problème de fabrication/reproductibilité
		Perf.	• Très grande capacité de stockage des interconnexions • Possibilité de vitesse élevée en mode statique (pas d'apprentissage)	• Mise à jour du cristal photoréfractif très lente • Efficacité diffractionnelle très faible

CHAPITRE III RÉSEAUX DE NEURONES ÉLECTRONIQUES

Le chapitre précédent a décrit les principales architectures optiques pour la réalisation de réseaux de neurones artificiels. Dans ce chapitre, nous décrivons les réalisations qui font appel à une autre technologie bien connue, l'électronique. Il existe deux approches courantes en électronique, l'approche numérique et l'approche analogique. De ces approches sont nées deux écoles de pensée où les partisans respectifs défendent leur position avec tenacité. Notre analyse tentera, autant que possible, de décrire de façon neutre chacune des approches en faisant ressortir leurs avantages et leurs inconvénients. Notre objectif n'est certes pas d'être en mesure de dire que l'approche *X* est meilleure que l'approche *Y*, mais plutôt d'affirmer que dans telles et telles situations, l'approche *X* est plus *avantageuse* que l'approche *Y*.

La section 3.1 traite d'abord de la complexité dans les réseaux de neurones électroniques. Cette analyse nous permettra de mieux comprendre les tendances dans les choix de conception des architectures. Par après, nous décrivons les architectures numériques dans la section 3.2. Finalement, les architectures analogiques sont décrites dans la section 3.3.

3.1 Complexité dans les réseaux de neurones électroniques

Pour réaliser un réseau de neurones dans une technologie donnée, le concepteur doit se poser une question fondamentale: "Comment réaliser de manière *efficace* un multiplieur

P153385.PDF [Page: 89 of 191]

vecteur-matrice?". En effet, l'essentiel du traitement d'un réseau de neurones se résume à cette simple équation algébrique[1]. Quoique simple, cette équation requiert un effort de calcul considérable lorsque les dimensions de la matrice sont élevées. Prenons comme exemple un vecteur de longueur N et une matrice de dimensions $N \times N$. Il s'agit là de l'exemple classique du réseau à interconnexion totale de type Hopfield. Le calcul d'un produit vecteur-matrice implique N^2 multiplications et $N(N-1)$ additions.

Le terme «efficace» dans notre question fondamentale revêt un sens très large et dépend évidemment des critères de tous et chacun. Règle générale, deux critères importants prévalent: le coût et la vitesse. En ITGE, le coût est généralement synonyme de surface de silicium utilisée. Idéalement, nous chercherons toujours à minimiser le coût et le temps de traitement. Si nous dénotons la surface par A et le temps de traitement par T, minimiser A et T simultanément revient à minimiser le produit AT. Ce produit est donc une métrique de l'efficacité d'un circuit ITGE.

Il existe une foule d'approches différentes pour réaliser un multiplieur vecteur-matrice, chacune d'elles étant caractérisée par sa métrique AT. Nous proposons d'examiner, en premier lieu, un sous-ensemble de ces approches: celles pour lesquelles les N^2 multiplications sont effectuées en parallèle. La figure 3.1 illustre un exemple. Les boîtes représentent des synapses (un multiplieur et son poids synaptique correspondant) et les cercles représentent des sommateurs.

Considérons tout d'abord l'approche de la figure 3.1. Sa complexité en surface est clairement déterminée par la surface occupée par les paires synapse-sommateur. La surface totale occupée est donc $A = O(N^2)$. Quant au temps de traitement, il est déterminé par le chemin qui parcourt le plus de sommateurs, puisque tous les produits sont calculés

1. Si le réseau fonctionne en mode performance (i.e. pas d'apprentissage)

P153385.PDF [Page: 90 of 191]

en parallèle. Comme un tel chemin parcourt N sommateurs, $T = O(N)$. Ainsi, le produit AT est donc $AT = O(N^3)$.

Pour que cette analyse de complexité soit valide, une hypothèse importante doit être émise: le délai des sommateurs doit être de beaucoup supérieur au délai imputable aux interconnexions. Pour une réalisation numérique, cette hypothèse est presque toujours rencontrée. Cependant, pour une réalisation analogique, si les signaux de sortie des synapses sont des courants, les sommateurs de la figure 3.1 sont tout simplement des nœuds. L'opération d'addition s'effectue donc instantanément, selon la loi de Kirchoff. Il apparaît alors clairement que les délais imputables aux interconnexions ne peuvent plus être négligés. Le délai à travers un fil est alors proportionnel au carré de sa longueur. Cependant, pour que ce délai soit comparable à celui à travers un multiplieur, il faut que la longueur du fil soit très grande. Ainsi, un réseau analogique pourrait avoir une métrique $AT = O(N^2)$ pour des N faibles, mais cette dernière pourrait atteindre $O(N^4)$ si N est très grand.

Pour les réalisations numériques, il existe une topologie plus astucieuse permettant de réduire le temps de traitement de $O(N)$ à $O(\log_2 N)$ au détriment d'une augmentation de surface d'$O(\log_2 N)$. Cette topologie est illustrée à la figure 3.2a. À présent, tous les

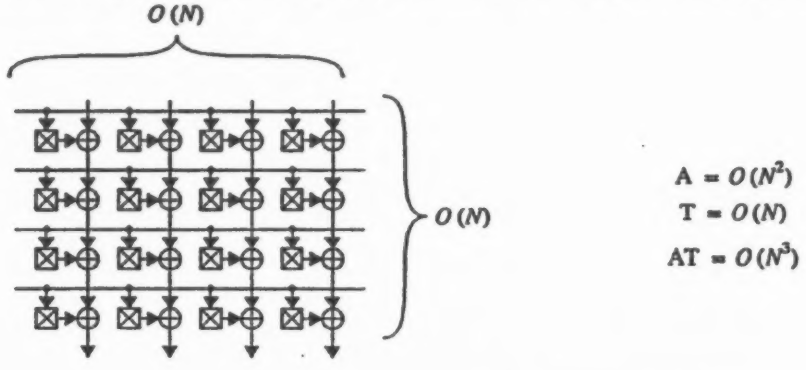


Figure 3.1: Topologie d'interconnexion de complexité $O(N^3)$.

P153385.PDF [Page: 91 of 191]

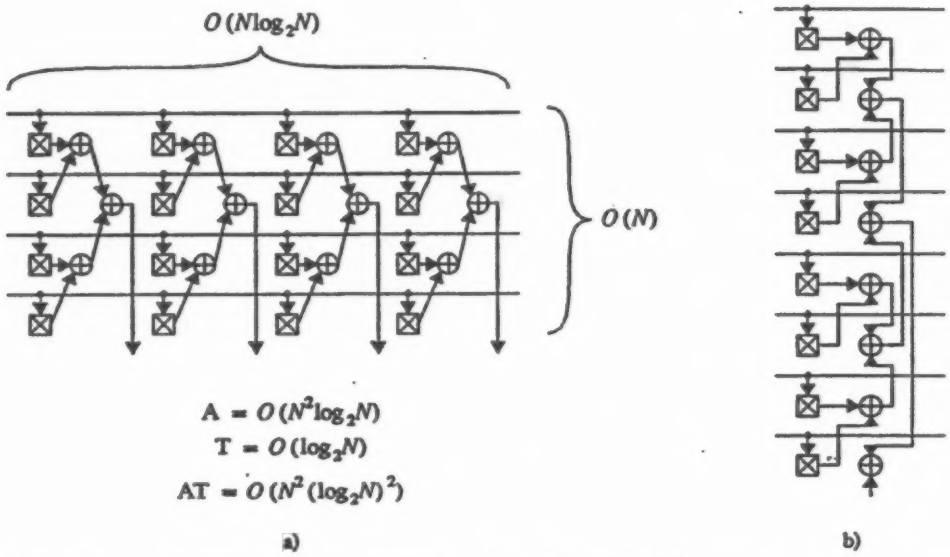


Figure 3.2: Topologie numérique de complexité $O(N^2 (\log_2 N)^2)$: a) Schéma d'interconnexion; b) Arbre de sommateurs replié: la complexité est déterminée par les fils.

chemins critiques sont équivalents et passent par $\log_2 N$ niveaux de sommateurs, d'où $T = O(\log_2 N)$. D'après la figure, l'augmentation de surface semble être la conséquence du nombre croissant d'étages de sommateurs, mais en réalité, ce n'est pas le cas. Le lecteur avisé aura remarqué qu'il est possible de replier l'arbre de sommateurs de sorte que chaque sommateur soit adjacent à une synapse, tel qu'illustré à la figure 3.2b [73]. Le plus grand nombre de fils traversant une droite imaginaire horizontale est $O(\log_2 N)$. C'est donc là la raison de l'augmentation de surface. Le produit AT est donc $AT = O(N^2 (\log_2 N)^2)$, ce qui représente une amélioration de $N/(\log_2 N)^2$ par rapport à la topologie de la figure 3.1.

Les expressions AT que nous venons de calculer sont toujours valides pour des N très grands. En pratique, N n'est jamais très grand, particulièrement en électronique numérique. En effet, le rendement d'un circuit est inversement proportionnel à sa surface, ce qui impose une limite pratique à la taille des circuits. Pour la topologie de la figure 3.2, le fac-

P153385.PDF [Page: 92 of 191]

teur $\log_2 N$ en surface commence à se manifester lorsque la largeur du réseau d'interconnexion des sommateurs devient comparable à la largeur d'une synapse. Pour un circuit numérique, il arrive souvent que les limites de taille imposées par le rendement soient atteintes avant même que cette situation ne se produise. La raison est que les multiplieurs numériques, occupent de très grandes surfaces, en particulier s'ils sont parallèles. La complexité du circuit est alors $A = O(N^2)$ et $AT = O(N^2 \log_2 N)$.

Jusqu'à date, nous avons supposé qu'autant pour l'analogique que pour le numérique, il était préférable d'effectuer les N^2 multiplications en parallèle. Cette supposition s'est avérée correcte pour ce qui est des architectures analogiques. En effet, comme nous le verrons dans la section 3.3, la plupart des puces analogiques développées ont adopté la topologie de la figure 3.1. En revanche, très peu de puces numériques ont adopté la topologie 3.2. La solution préconisée consiste à calculer, de façon semi-parallèle, N connexions à chaque cycle par une rangée de N processeurs (multiplieurs-accumulateurs). Le calcul total des N^2 connexions est donc terminé après N cycles, et le produit AT est donc $O(N^2)$. En fait, nous avons négligé un facteur important qui contribue à augmenter la surface: la mémoire. Dans la topologie 3.2, la mémoire est distribuée également entre les synapses: chaque synapse a accès à son poids. En réduisant le nombre de synapses à N, la mémoire doit donc être regroupée en N modules de N poids chacun, ce qui contribue à augmenter la surface de la puce. Cette augmentation compense quelque peu pour la différence de $\log_2 N$ entre les métriques AT des deux topologies, de sorte qu'on puisse à toutes fins pratiques les considérer équivalentes (du point de vue AT).

Bien qu'elles soient équivalentes du point de vue AT, les deux topologies ne le sont certes pas du point de vue pratique. Nous avons déjà mentionné qu'étant donné les dimensions élevées des synapses numériques, un petit nombre d'elles peuvent être intégrées sur chaque puce. La mémoire d'un tel système, si chaque synapse comporte un seul poids, est

P153385.PDF [Page: 93 of 191]

relativement faible, et conséquemment, seuls les réseaux de faible taille pourront être émulés. Pour émuler de plus grands réseaux, une nouvelle matrice synaptique devrait périodiquement être chargée dans la puce. Comme il est peu réaliste d'imaginer un port pour chaque synapse (une telle approche nécessite $O(N^3)$ en surface et $O(N^2)$ en ports), ce chargement devrait sans doute se faire sériellement, ce qui ferait chuter les performances de façon très appréciable. En revanche, la topologie semi-parallèle ne souffre pas de ces limitations. En effet, tout l'espace sauvé en éliminant $(N-1) \times N$ synapses peut être utilisé pour intégrer des mémoires très denses. Par conséquent, de très grand réseaux peuvent être émulés.

Pour les architectures analogiques, étant donné les très faibles dimensions des synapses (voir 3.3.2), un très grand nombre d'elles peuvent être intégrées et des réseaux de taille acceptable peuvent alors être émulés. N'oublions pas que la nature analogique des signaux permet d'utiliser une seule broche physique par signal, contrairement aux réalisations numériques qui doivent utiliser b broches par signal, où b est la résolution des signaux.

Nous entreprenons donc notre analyse des réseaux de neurones électroniques par les architectures numériques. Suivra après l'analyse des architectures analogiques en 3.3.

3.2 Architectures numériques

Depuis l'avènement des premiers ordinateurs au début des années soixante, la représentation numérique de l'information est devenue pratique courante pour un nombre toujours croissant d'applications. De nos jours, même les applications pour lesquelles l'analogique était maître il y a environ dix ans se réorientent progressivement vers le numérique. Quelques exemples d'actualité sont la téléphonie, l'audiophilie, la télédiffusion à haute définition. Nous pouvons attribuer cette immense popularité à deux facteurs principaux: l'immunité au bruit ainsi que l'aisance de programmation que procure cette représentation

P153385.PDF [Page: 94 of 191]

(que nous appellerons *programmabilité*). En effet, il est possible d'obtenir un rapport signal sur bruit de quantification arbitrairement élevé pour un signal donné, simplement en augmentant le nombre de bits qui le représente. La programmabilité, quant à elle, est due à la modularité des constituants principaux des architectures numériques: les portes logiques. En agençant quelques portes logiques de base (ET, OU, OU exclusif, etc.), il est possible d'obtenir une foule d'organes de calcul numériques (sommateurs, multiplieurs, détecteurs de parité, etc.).

Le choix en faveur d'une mise en œuvre électronique numérique ou analogique est loin d'être évident, ce que nous tenterons de faire réaliser au lecteur. Une chose est certaine: l'utilisation d'une technologie sera toujours justifiée si l'application quelle doit servir profite au maximum des avantages et des bienfaits de cette technologie. Dans ce qui suit, nous allons décrire quelques-unes des architectures numériques les plus populaires dans le but de faire ressortir cette vérité.

3.2.1 Principe

Une architecture est dite *numérique* si l'information qu'elle traite est numérisée, c'est-à-dire quantifiée en amplitude. Dans la plupart des cas, la base 2 (le *binaire*) est utilisée pour représenter cette information. Le choix de cette base permet aux concepteurs de RNA de recycler toute l'expertise accumulée au fil des années en conception de circuits numériques binaires (multiplieurs, sommateurs, circuit de propagation de la retenue, etc.).

Les concepteurs de RNA numériques ont le choix parmi trois grandes approches de conception [2]. Ces approches sont:

- Simulation d'algorithmes neuronaux sur des machines parallèles à usage général;
- Conception de neuro-ordinateurs (cartes spécialisées comportant des processeurs à usages généraux comme le M68030 ou le TMS320C40);
- Conception de puces neuronales dédiées.

P153385.PDF [Page: 95 of 191]

Selon l'approche choisie, une foule d'autres choix de conception s'imposent. Par exemple, pour la première approche, le concepteur doit choisir parmi les divers types d'ordinateurs parallèles (Transputers, machine WARP, etc.). De plus, un algorithme de correspondance entre les neurones et les processeurs doit être déterminé. Bien souvent, ce choix déterminera les performances globales du système.

La conception de neuro-ordinateurs comporte elle aussi ses choix de conception. Mentionnons le choix des processeurs (M68030, TMS320C40, etc.), la taille de la mémoire locale aux processeurs, le nombre de processeurs et la précision des calculs.

Par opposition à des mises en œuvre basées sur l'emploi de processeurs existants, l'approche de choix durant les dernières années a été la conception de puces neuronales dédiées. Ce penchant est en partie dû à l'accessibilité grandissante des technologies d'intégration numériques. Notre étude se limitera donc essentiellement à cette approche. Ceci peut, à première vue, paraître restrictif. Cependant, nous aurons l'occasion de discuter de la mise à l'échelle de systèmes constitués de plusieurs puces dédiées. Les systèmes ainsi conçus s'apparentent alors à la deuxième classe, celle des neuro-ordinateurs, avec comme seule différence le fait que les constituants sont des puces dédiées plutôt que des processeurs à usages généraux commercialement disponibles.

La conclusion importante de la section précédente est que le parallélisme complet des interconnexions est rarement utilisé dans les architectures numériques. Dans ce qui suit, nous allons décrire les deux principaux choix de conception pour les RNA numériques, tous deux associés au compromis performance-parallélisme. Ces choix de conception sont:

- le multiplexage temporel des ressources de communication;
- le recours au traitement systolique.

P153385.PDF [Page: 96 of 191]

Multiplexage temporel des ressources de communication

La topologie d'interconnexion la plus efficace, lorsque le degré d'interconnexion requis par une application est élevé (comme c'est le cas pour les RNA), est sans conteste le bus de diffusion (broadcast bus) [36]. Dans cette topologie, n processeurs partagent deux bus: un d'entrée et un autre de sortie. La figure 3.3 illustre un exemple d'architecture à bus de diffusion, l'architecture CNAPS (*Connected Network of Adaptive ProcessorS*) proposée par Hammerstrom. Chaque puce comporte 64 processeurs élémentaires qui fonctionnent selon le principe de traitement SIMD (Single Instruction, Multiple Data). Ainsi, pour calculer N^2 connexions, si un processeur élémentaire est assigné à chaque neurone (pour un total de N processeurs), N cycles sont nécessaires. Notez que cette architecture est un exemple de l'architecture semi-parallèle que nous avons décrite à la section 3.1.

Recours au traitement systolique

Le recours au traitement systolique est une technique qui a connu beaucoup de succès dans les applications de traitement numérique des signaux. Les architectures systoliques ont la qualité de maximiser la performance (vitesse) tout en minimisant la bande passante requise pour l'entrée des données et leur sortie. Elles sont particulièrement bien adaptées au calcul matriciel, une part importante des calculs dans un RNA.

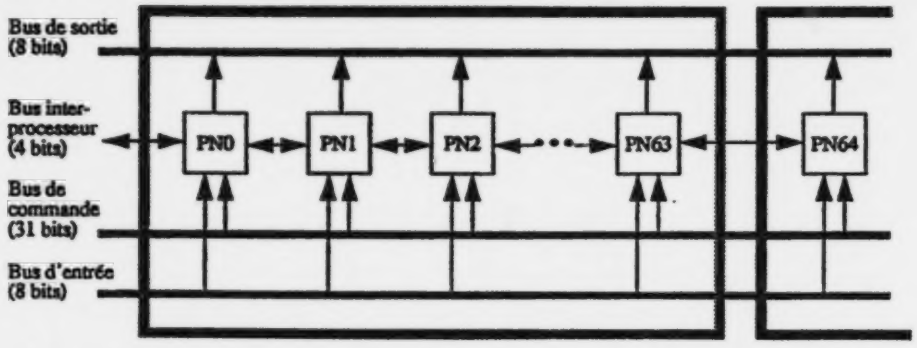


Figure 3.3: Schéma de l'architecture CNAPS de Hammerstrom. Le cadre gris représente les frontières d'une puce.

P153385.PDF [Page: 97 of 191]

Dans [48] et [51], une architecture systolique circulaire (ring systolic) est proposée. Elle est constituée d'une chaîne de processeurs élémentaires communiquant par un lien unidirectionnel avec leur voisin de gauche (voir figure 3.4). De plus, le processeur à l'extrême gauche communique avec le processeur de l'extrême droite pour ainsi former une boucle (d'où le nom *circulaire*). Chaque processeur est responsable de calculer l'entrant net du neurone qu'il émule. Cet entrant net, défini à l'équation (1.1), est:

$$t_j = \sum_{i=1}^{N} x_i w_{ij}, \tag{3.1}$$

ou sous forme matricielle:

$$\mathbf{T} = \mathbf{XW}. \tag{3.2}$$

Pour calculer son entrant net, chaque neurone j a besoin des N poids $\{w_{ij}\}$ qui lui sont reliés, ainsi que les N valeurs d'activation s_k des autres neurones. Dans l'architecture systolique circulaire, les poids sont acheminés aux processeurs par le bas et les activations se propagent vers la gauche. À chaque cycle, N poids sont présentés aux N processeurs (1 poids par processeur). Pour synchroniser l'arrivée des poids avec les valeurs d'activation

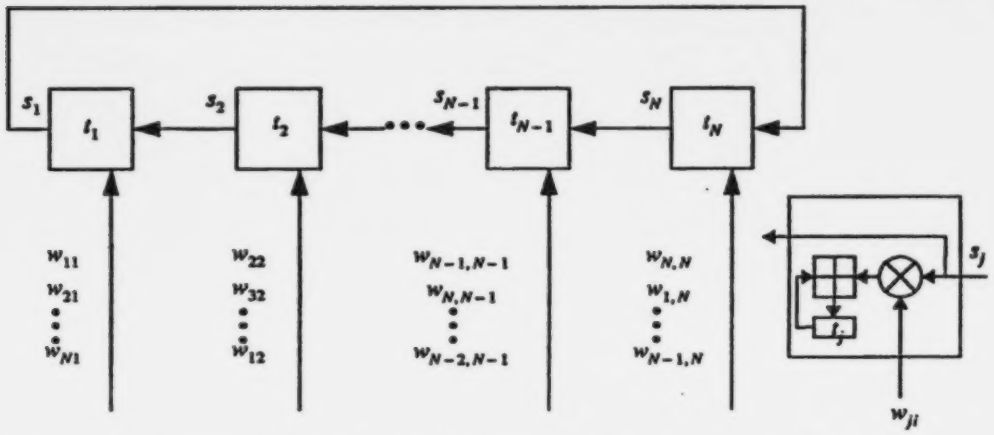


Figure 3.4: Schéma d'une architecture systolique circulaire (d'après [51]).

P153385.PDF [Page: 98 of 191]

appropriées, les colonnes de la matrice de poids sont décalées circulairement. Ainsi, la $j^{ème}$ colonne sera décalée de $j-1$ éléments vers le bas. La figure 3.4 clarifie le processus.

3.2.2 Considérations d'ordre pratique

Nous avons déjà mentionné que le multiplexage des communications par diffusion permet d'échanger de la surface de silicium contre de la vitesse. Outre les économies de surface, cette technique permet également de résoudre (en partie) les problèmes de bande passante. Seules les broches pour les quatre bus et l'alimentation sont nécessaires. Bien entendu, les processeurs devront avoir leur mémoire locale, sinon les besoins en bande passante limiteront sévèrement le nombre de processeurs qu'il est possible d'intégrer par puce. Dans l'architecture CNAPS, chaque processeur dispose d'une mémoire locale de 4Ko, ce qui limite le nombre de neurones à 2048 si on désire 2048^2 interconnexions à 16 bits de précision chaque (2048 processeurs seront nécessaires). D'où l'avantage de l'expansibilité que procure cette architecture (voir figure 3.3). Plusieurs puces CNAPS peuvent être cascadées simplement en reliant les 4 bus de façon parallèle.

Le problème avec les architectures à bus de diffusion est que lorsque l'interconnexion n'est pas totale, les performances commencent à chuter. La raison est que lorsqu'un PE (Processeur Élémentaire) diffuse son état aux autres processeurs (en reliant le bus de sortie au bus d'entrée), ce ne sont pas tous les processeurs qui «écoutent». Un autre problème existe en ce qui a trait à la tolérance aux pannes. Lorsqu'un lien physique existe pour chaque interconnexion, la defectuosité de quelques liens n'est pas catastrophique. En effet, la nature adaptative des réseaux de neurones permettra d'absorber ces pannes, sans avoir d'impact majeur sur la performance. Cependant, une panne dans un des bus de diffusion paralyse tous les n processeurs. Des techniques de signalisation tolérantes aux pannes sont proposées dans [3] mais elles exigent la redondance physique des liens de communication.

P153385.PDF [Page: 99 of 191]

Les architectures faisant appel au traitement systolique offrent sensiblement les mêmes compromis surface-vitesse que les architectures à bus de diffusion. En effet, il est clair d'après la figure 3.4 qu'après N cycles, N^2 interconnexions auront été calculées.

Tout comme l'architecture CNAPS, l'architecture systolique souffrira des mêmes problèmes de bande passante en ce qui concerne les poids. Le degré de parallélisme, et par conséquent le nombre de processeurs sera déterminé par la bande passante de la puce, qui est elle même déterminée par le nombre restreint de broches sur la périphérie de la puce. Bien entendu, si la puce comporte une mémoire locale, comme c'est le cas pour l'architecture CNAPS, le problème de bande passante est à toutes fins pratiques éliminé. Ces architectures pourraient sans aucun doute profiter de la technologie d'intégration sur plaquette (Wafer Scale Integration) pour avoir accès à une vaste quantité de mémoire.

Il n'y a pas que les contraintes de bande passante qui limitent le nombre de processeurs que l'on puisse intégrer par puce. En effet, lorsque la surface de la puce est très grande, des problèmes de synchronisation d'horloge (*clock skew*) vont indubitablement se manifester. Il devient alors potentiellement utile d'adopter des techniques d'auto-synchronisation (*self-timed*) pour remédier au problème. Les techniques d'auto-synchronisation sont souvent coûteuses en raison de la duplication des canaux de communication et de la logique additionnelle nécessaire pour l'établissement du protocole d'auto-synchronisation. Notez que cette restriction s'applique aussi bien aux architectures systoliques qu'aux architectures à bus de diffusion.

3.2.3 Performances

Dans cette section, nous compilons les données de performance de quelques-unes des puces dédiées développées par divers chercheurs au cours des dernières années. Comme le nombre de neurones simulables n'est en général pas une contrainte pour les architectures

P153385.PDF [Page: 100 of 191]

numériques, nous nous contenterons de compiler les données de vitesse (en connexions par seconde). La raison est que, pour une architecture numérique, il est toujours possible d'utiliser un seul et même processeur pour traiter (séquentiellement) plusieurs neurones. On parle alors de mise en œuvre *virtuelle*, puisqu'il est impossible d'identifier physiquement les neurones sur le circuit. Pour les architectures analogiques, comme nous le verrons dans la section 3.3.3, la situation est différente.

Bien entendu, tous les neurones virtuels assignés à un neurone physique sont traités de façon séquentielle. Par conséquent, théoriquement, un nombre infini de neurones peuvent être simulés, si le temps le permet. En réalité, le nombre de neurones sera limité par la mémoire du système. Ceci peut être déterminant dans le cas d'un système qui tire sa performance du fait qu'il a accès à une mémoire interne. Cependant, pour les processeurs qui utilisent de la mémoire externe, comme le MA16 de Siemens que nous décrirons sous peu, cette dernière peut à toutes fins pratiques être arbitrairement augmentée (les stations de travail de nos jours peuvent avoir accès à 64 Mo de mémoire vive).

Lors de l'interprétation des données, une mise en garde s'impose. En effet, ces dernières supposent toujours la situation idéale qui permet d'obtenir le débit le plus élevé. Des irrégularités dans la matrice synaptique peuvent diminuer considérablement ce débit, comme nous l'avons décrit pour le cas de l'architecture CNAPS. L'impact d'une telle irrégularité dépend fortement de sa nature et de l'architecture utilisée. Son analyse est donc trop complexe pour être discutée ici.

Notre compilation débute avec l'architecture neuro-WSI (*Wafer Scale Integration*), telle que décrite dans [81]. Cette dernière fait appel à la diffusion pour le multiplexage des communications ainsi qu'à l'intégration sur plaquette pour maximiser le nombre de neurones. Elle est constituée de 576 neurones numériques. La précision est de 9 bits pour les

P153385.PDF [Page: 101 of 191]

sorties des neurones et 8 bits pour les poids. Chaque neurone dispose d'un fichier de registre rapide faisant office de mémoire locale. Cette mémoire a la particularité de ne retenir que les 64 plus grands poids (i.e. les poids qui auront la plus grande influence sur le neurone). Une adresse identifiant le neurone d'origine est associée à chaque poids. Ainsi, à chaque cycle, le neurone émetteur diffuse sa valeur d'activation et son adresse à tous les autres neurones. Ces derniers vérifient alors s'ils possèdent un lien avec le neurone émetteur (i.e. s'ils possèdent un poids). Si c'est le cas, ces derniers accumulent le produit synaptique à l'aide d'un sommateur accumulateur. Sinon, une valeur nulle est accumulée.

Le neuro-WSI effectue une diffusion à toutes les 464 ns. Par conséquent, les 576 diffusions ont lieu en un total de 267 µs. Le nombre effectif de connexions synaptiques réalisées est de $576 \times 64 = 36864$ (et non 576×576), pour une vitesse de $36864\ \mathrm{C} \div 267\mu\mathrm{s} \approx 140$ MCS.

L'architecture CNAPS de Hammerstrom comporte 64 processeurs avec 4096 octets de mémoire pour un total de 256Ko par puce. La société Adaptive Solutions fabrique des cartes comportant jusqu'à 4 puces (256 proceseurs). Un système comportant deux cartes de quatre puces chaque est également disponible. La puce peut opérer en trois modes de précision pour les poids: 1 bit, 8 bits ou 16 bits. N connexions (multiplication-accumulation) de 8 ou 16 bits peuvent être exécutées en un seul cycle. Pour les poids de 1 bits, $N \times 8$ connexions sont exécutées en parallèle. La fréquence de l'horlode est de 20MHz. Donc, pour des poids de 8 ou 16 bits, la vitesse du système varie de $64 \times 20{\times}10^6 = 1.28$ GCS à $512 \times 20{\times}10^6 = 10.24$ GCS pour les configurations de 1 à 8 puces respectivement. Ces performances sont multipliées par un facteur 8 si des poids binaires sont utilisés. Le nombre de neurones n'est pas limité au nombre de processeurs. En effet, plusieurs neurones peuvent être émulés par un même processeur. Par exemple, avec des poids de 8 bits, et une configuration de 8 puces (512 processeurs), nous avons à notre disposition une possibilité

P153385.PDF [Page: 102 of 191]

de 2097152 interconnexions. Avec une telle capacité, il serait possible de réaliser 8 couches de 512 neurones (pour un total de 4096 neurones) où chaque couche serait complètement interconnectée avec la suivante. Donc, pour des poids de 8 bits, chaque puce peut émuler jusqu'à 512 neurones (8 couches × 64). Il est cependant important de ne pas perdre de vue que les connexions ne seront réalisées que *N* à la fois.

Tout récemment, la compagnie Siemens à développé une puce neuronique dédiée qui a pour nom MA16 [69]. Cette dernière est constituée de quatre processeurs numériques ultra-rapides. Ces processeurs ont pour fonction d'effectuer, selon leur mode de programmation (dicté par un contrôleur local), toutes les primitives de calcul intrinsèques aux algorithmes neuronaux. Ces primitives sont la mutliplication matrice-matrice, l'addition et la soustraction de matrices, la transposition de matrices, le calcul de normes ainsi que la recherche de l'élément maximum ou minimum d'une matrice. Chaque processeur fait appel au traitement systolique pour calculer les primitives. Une chaîne linéaire de 4 multiplieurs-accumulateurs constitue le cœur du processeur. Chaque multiplieur est entièrement parallèle et fait appel à la technique «carry-save», ce qui explique que l'essentiel de la surface de la puce est résevée aux multiplieurs. Le MA16 ne comporte pas de mémoire locale autre que les tampons d'entrée et de sortie. Cependant, la vitesse élevée (50MHz dans le pire cas) et le nombre élevé de broches (319) permettent tout de même un débit total de 10.9 Gb/s. Les auteurs ont mesuré une vitesse de 800 MCS (une connexion=16 bits) par puce. Une matrice 16x16 de puces MA16 pourrait même atteindre des performances allant jusqu'à 200 GCS [70].

La compagnie *Neural Technologies* d'Angleterre offre une gamme de cartes accélératrices pour les PC. Ces cartes ont toutes la particularité de comporter des modules d'acquisition de données (qu'elles soient de nature audio, vidéo, etc.), ce qui contribue grandement à accélérer le débit total du système. Le cœur de la carte NT5000 est constitué

P153385.PDF [Page: 103 of 191]

d'un microcontrôleur 8 bits opérant à une fréquence de 16 MHz et d'un coprocesseur 32 bits à 10 MIPS. La version turbo de la NT5000 comporte une mémoire locale de 256 Ko et est capable d'opérer à un débit de 2 MCS. L'apprentissage du réseau de neurone prend place dans le PC. L'accélérateur agit alors uniquement à titre de carte d'acquisition de données. Une fois la phase d'apprentissage terminée, le réseau de neurones est chargé dans la mémoire locale de l'accélérateur et le PC devient alors un simple moniteur.

La société Neural Technologies distribue également une puce spécialisée, le NiSP (Neural Instruction Set Processor). Le NiSP est un processeur sériel qui comporte un multiplieur de 16 bits et un accumulateur de 35 bits. La fréquence d'opération est de 40 MHz, ce qui se traduit par un débit de 40 MCS. Le NiSP est capable de supporter n'importe quelle topologie multicouche de RNA comportant un maximum de 65536 neurones et synapses.

Finalement, mentionnons la plus récente puce d'Intel, le Ni1000. Contrairement à son prédécesseur (le 80170NX décrit à la section 3.3.3), le Ni1000 est plutôt un accélérateur numérique pour la reconnaissance de formes qu'un réseau de neurones. Sa fonction est de calculer les similarités entre un patron d'entrée et 1024 prototypes stockés en mémoire. Les patrons d'entrée comportent 256 éléments de 5 bits chaque. Les 1024 prototypes sont stockés dans une mémoire FLASH EEPROM. À une fréquence d'horloge de 40 MHz, le Ni1000 est capable de procurer un débit de 40 000 classifications à la seconde. Un réseau de neurones accomplissant une tâche similaire comporterait $1024 \times 256 = 262144$ synapses, et devrait opérer à une vitesse de $262144 \times 40000 = 10$ GCS. Ce chiffre confirme que moins un processeur est général, plus son débit est élevé.

Le tableau 3.1 compile les vitesses des diverses architectures que nous venons de décrire. La vitesse de deux systèmes, l'un comportant 4 puces CNAPS, l'autre comportant

P153385.PDF [Page: 104 of 191]

4 puces MA16 sont listées dans la rangée «vitesse système». Les cinq dernières architectures (WARP, MarkIII et IV, Delta et Anza Plus) sont décrites dans la référence [2].

Tableau 3.1: Vitesses de diverses architectures de RNA numériques.

Caractéristiques		neuro-WSI [51]	Adaptive Solutions CNAPS	Siemens MA16	Neural Technologies NT5000	Neural Technologies NISP	NI1000	WARP	TRW Mark III	TRW Mark IV	SAIC Delta	HNC Anza Plus
Vitesse (en MCS)		140	1280	800	2	40	10K	17	0.45	5	10	6
Vitesse système (en GCS)			5.12	3.2								
Approche	Machine parallèle							✓				
	Neuro-ordinateur		✓	✓	✓				✓	✓	✓	✓
	Puce neuronale dédiée	✓	✓	✓		✓	✓					
Disponibilité commerciale			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Il importe de mettre en garde le lecteur contre une interprétation trop hative des données du tableau 3.1. En effet, les architectures décrites diffèrent sur plusieurs facteurs, les

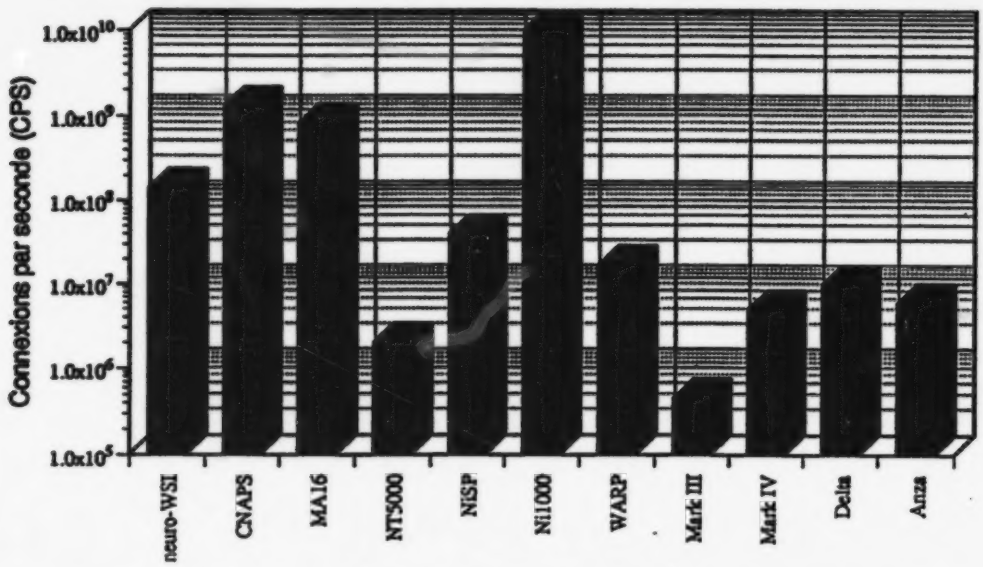


Figure 3.5: Graphique des performances des architectures numériques décrites dans le texte.

P153385.PDF [Page: 105 of 191]

plus importants étant leur flexibilité et leur exactitude. Ainsi, certaines architectures ne peuvent émuler qu'un nombre très restreint d'algorithmes neuronaux alors que d'autres effectuent des calculs avec une très faible précision. Les données du tableau 3.1 y figurent donc simplement à titre d'information, dans le but d'apprécier les ordres de grandeur. Lors du choix d'une architecture pour une éventuelle acquisition, une étude plus approfondie des particularités de l'architecture et de l'application s'impose. Les mêmes conclusions s'appliqueront pour les architectures analogiques, que nous décrivons dans la section suivante.

3.3 Architectures analogiques

L'électronique analogique est la deuxième approche que nous considérons pour la mise en œuvre de réseaux de neurones électroniques. Cette approche tire son originalité du fait qu'elle s'apparente davantage aux réseaux de neurones biologiques que l'approche numérique. Son principal attrait est la facilité avec laquelle certaines primitives de calcul peuvent être réalisées. Par exemple, si deux signaux sont représentés par des courants, leur somme est obtenue par la loi de Kirchoff, en connectant les deux fils qui portent ces signaux. De plus, les organes de calcul, comme les multiplieurs par exemple, comportent en général moins de transistors que l'organe équivalente en numérique, ce qui permet d'obtenir un niveau d'intégration supérieur en analogique. Commençons notre analyse des architectures analogiques par décrire en premier lieu leur principe.

3.3.1 Principe

Une architecture est dite *analogique* si l'information qu'elle traite est sous forme de courant ou de tension analogique. Contrairement à bon nombre d'architectures numériques qui multiplexent leurs ressources pour économiser de la surface, les architectures analogiques utilisent des organes de calculs dédiés pour chaque neurone et chaque inter-

P153385.PDF [Page: 106 of 191]

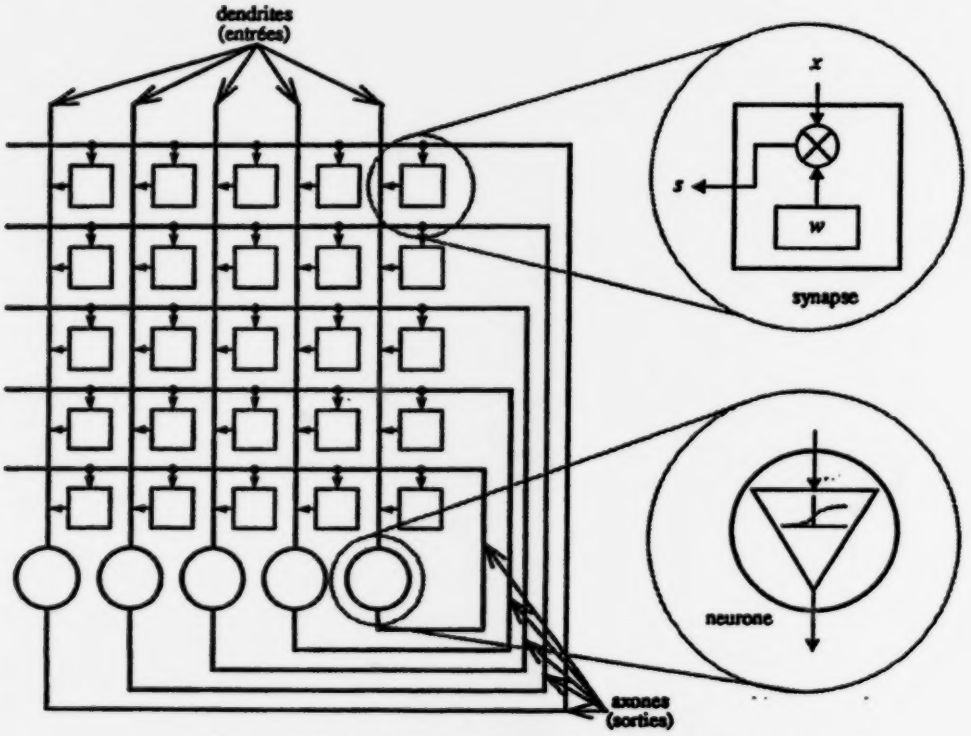


Figure 3.6: Schéma type d'une architecture analogique.

connexion, en raison de leurs faibles dimensions. La figure 3.6 illustre un exemple type d'un réseau de neurone analogique.

Tout le défi dans la conception d'une architecture analogique réside dans la conception des synapses. En effet, la conception des neurones est relativement simple, car une fonction de seuillage se réalise très bien en électronique analogique. En revanche, la circuiterie des synapses est plus complexe. De plus, pour un circuit qui comporte N neurones, un total de N^2 synapses sera requis pour une interconnexion totale. Il est donc clair que la majeure partie de la surface d'une puce sera allouée aux synapses, d'où l'importance de bien les optimiser.

P153385.PDF [Page: 107 of 191]

Le concepteur de RNA analogiques devra aussi choisir judicieusement la nature (courant ou tension) des différents signaux de son circuit. Par exemple, plusieurs concepteurs ont choisi de représenter les signaux de sortie des neurones par des tensions et leurs signaux d'entrée par des courants. L'art dans la conception réside dans le choix qui minimisera la dimension des circuits ou qui améliorera les performances. Ainsi, le choix de représenter les signaux d'entrée d'un neurone par des courants est intelligent car le neurone devra additionner tous ces signaux ensemble. Additionner des courants est très simple à réaliser (loi de Kirchoff), tandis qu'additionner des tensions est plus compliqué. D'autre part, distribuer une tension à plusieurs cellules est simple (il s'agit tout simplement de court-circuiter les entrées des cellules avec le signal à distribuer), tandis que distribuer un courant est légèrement plus compliqué, mais très encombrant (utilisation de miroirs de courant).

3.3.2 Considérations d'ordre pratique

La plupart des problèmes de design des architectures analogiques sont rencontrés lors de la conception des synapses, comme nous l'avons mentionné dans la section précédente. Plus particulièrement, les problèmes portent sur la manière de stocker les poids d'interconnexion. Aussi, nous porterons une attention sur ce sujet. Contrairement à l'électronique numérique, où l'information est stockée sous forme binaire dont les niveaux sont regénérés automatiquement et de façon fiable, l'électronique analogique n'offre pas de moyens simples et peu coûteux pour regénérer le niveau d'une quantité analogique. Au cours des dernières années, plusieurs stratégies ont été proposées. Ces stratégies ont toutes la propriété d'échanger de la *surface de silicium* contre de la *résolution*. La résolution est le nombre de niveaux qu'il est possible de représenter fidèlement avec le médium. Elle est souvent exprimée en bits, dans le but de la comparer avec l'approche numérique. Sa valeur est alors le logarithme en base 2 du nombre de niveaux. Par exemple, un médium

P153385.PDF [Page: 108 of 191]

pouvant stocker une valeur avec un précision de 256 niveaux aura une résolution de 8 bits. Il va de soi que la résolution d'une synapse a un impact direct sur le niveau d'intégration d'un circuit neuronal analogique. Ainsi, plus la résolution des synapses est faible, plus la surface qu'ils occupent est faible et donc plus il sera possible d'en intégrer un grand nombre pour une surface de silicium donnée.

Horio et Nakamura [42] ont identifié quatre grandes classes de mémoires analogiques:

- Mémoires numériques contrôlant une valeur analogique quantifiée
- Condensateurs
- Mémoires Non-Volatiles (MNV)
- Mémoires analogiques actives (MAA)

3.3.2.1 Mémoires numériques contrôlant une valeur analogique quantifiée

Cette technique est un compromis entre les mémoires numériques et les mémoires analogiques. La valeur du poids est stockée sous forme binaire, dans un registre. Chaque bit du mot binaire commande une source de courant dont la «force» est proportionnelle au poids du bit correspondant, tel qu'illustré à la figure 3.7. Par exemple, le bit le moins significatif commandera une source de courant de force I_u, le bit suivant commandera une source de force $2I_u$, le suivant, $4I_u$, et ainsi de suite. Les courants sont alors combinés sur une même ligne pour produire le courant total résultant, lequel est proportionnel à la

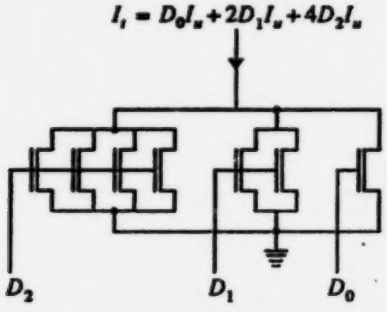


Figure 3.7: Génération d'un courant analogique à partir d'une valeur numérique.

P153385.PDF [Page: 109 of 191]

valeur du poids synaptique. La force des sources de courant est ajustée soit en augmentant la largeur W du canal du transistor agissant comme source de courant, soit en branchant plusieurs transistors en parallèle, diminuant ainsi l'impédance de la source. Il apparaît donc clairement que cette approche est restreinte à des résolutions d'au plus 3 ou 4 bits car la surface du circuit croît de manière exponentielle avec le nombre de bits de résolution. Il existe certaines techniques qui atténuent cette croissance exponentielle. Cependant, les variations de seuil entre les transistors (dues entre autres aux variations de concentration des dopants) sont une autre cause de limitation de la résolution.

3.3.2.2 Condensateurs

Un façon élégante de mémoriser la valeur d'un poids synaptique consiste à accumuler des charges électriques sur une des plaques d'un condensateur. La figure 3.8 illustre un montage possible. La tension V_w est placée sur le condensateur C_w à l'aide de la porte de transmission M_p. Lorsque le condensateur a atteint la tension voulue, le signal V_s à la grille de M_p bloque ce dernier et la charge sur C_w est gardée prisonnière. L'avantage de ce montage est que la charge sur C_w peut être échantillonnée par la grille du transistor M_s sans être dégradée. Le courant de drain de M_s sera alors modulé par la valeur du poids mémorisé.

Le principal inconvénient en défaveur de l'utilisation des condensateurs comme mémoire est leur faible *temps de rétention*. La rétention est la capacité pour un élément de mémoire de bien conserver sa charge. Cette faible rétention est principalement due aux

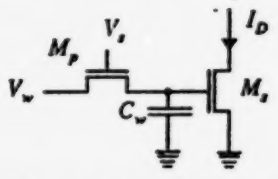


Figure 3.8: Stockage d'une tension analogique à l'aide d'un condensateur (adapté de [42]).

P153385.PDF [Page: 110 of 191]

courants de fuite qui tendent à décharger le condensateur. Ces courants de fuite peuvent être minimisés en augmentant la périphérie de la plaque du condensateur. Si l'épaisseur d de l'oxyde est maintenue constante, la capacité du condensateur sera donc augmentée puisque $C = \varepsilon A/d$, où A est la surface du condensateur et ε est la permittivité du diélectrique. De plus, l'opération d'écriture (dépôt de charges) sera plus longue car la constante de temps d'écriture RC sera plus grande, sans oublier que le condensateur lui-même occupera beaucoup plus de place. Mais la principale composante qui contribue à décharger le condensateur provient du courant infraseuil du transistor d'adressage M_p. En effet, même lorsque V_s est maintenu inférieur à V_{th}, un faible courant courant infraseuil circulera dans le canal de M_p, contribuant ainsi à décharger C_w (voir page 18).

Il est à noter que certains algorithmes neuronaux ne requièrent pas un temps de rétention élevé. Pour ces applications, il est donc possible d'intégrer beaucoup de synapses rapides sur une surface donnée. Dans certains cas, il est même possible d'utiliser le condensateur de grille d'un transistor comme élément de mémoire au quel cas la surface requise pour le stockage est réduite à néant.

Une autre technique plus ou moins populaire pour minimiser les courants de fuite consiste à refroidir le circuit. En effet, les courants de fuite décroissent de manière exponentielle avec la température [42].

Une manière astucieuse de compenser l'effet des courants de fuite est de stocker la valeur comme une différence entre deux tensions (sur deux condensateurs différents)[34]. Comme les courants de fuite tendent à décharger les condensateurs de façon quasi-identique, la différence de tension tend à demeurer constante. Cette différence peut ensuite être détectée et convertie en courant par un amplificateur de transconductance, tel qu'illustré à la figure 3.9. Pour des différences de tension faible et pour des variations de tension V_{in}

P153385.PDF [Page: 111 of 191]

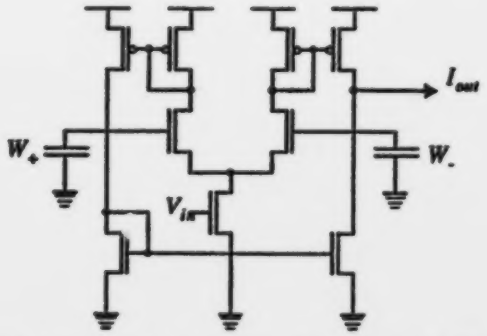


Figure 3.9: Synapse analogique à mémoire différentielle (d'après [34]).

faibles, le courant de sortie I_{out} est approximativement proportionnel au produit $V_{in}(W_+ - W_-)$ où W_+ et W_- sont les tensions aux bornes "+" et "-" de l'amplificateur différentiel.

3.3.2.3 Mémoires non-volatiles

Nous avons mentionné qu'un des inconvénients majeurs de la technique précédente était le faible temps de rétention des condensateurs utilisés. Ce problème peut être grandement réduit si des transistors dits «non-volatiles» sont utilisés pour stocker les poids. Il existe deux types de transistors non-volatiles: les transistors *à emprisonnement de charges* et les transistors *à grille flottante*. Tous deux fonctionnent à partir du même principe: la conductance du transistor est modifiée en variant le seuil du transistor en fonction du poids synaptique à stocker. La figure 3.10a illustre un exemple du premier type de transistors non-volatiles: le transistor MNOS (Metal-Nitride-Oxide Semiconductor). Pour programmer un poids, une impulsion de tension positive à voltage élevé est appliquée sur la grille. Ceci a pour effet d'attirer des électrons du canal jusqu'à la couche de nitrure (en passant par la couche d'oxyde) par effet tunnel[1], et donc d'augmenter le seuil du transistor V_s. Lorsque la tension de grille est retirée, les électrons se retrouvent emprisonnés dans la

1. L'effet tunnel porte le nom de «Fowler-Nordheim tunneling»

P153385.PDF [Page: 112 of 191]

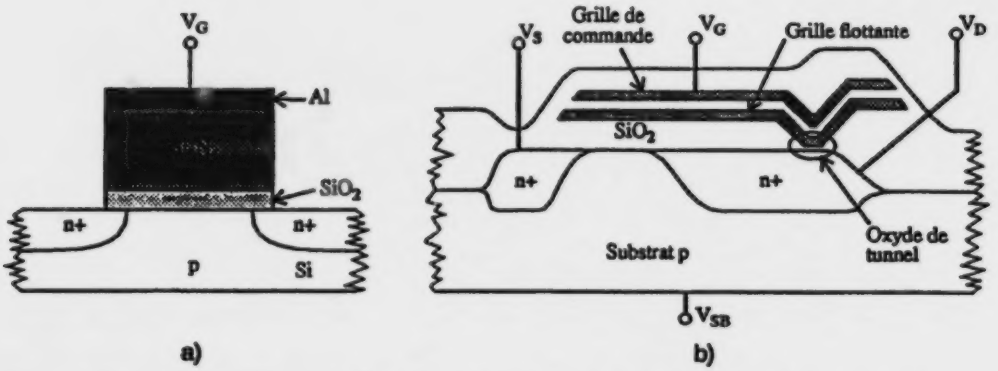


Figure 3.10: Vue en coupe a) d'un transistor MNOS à canal n (d'après [42]); b) d'un transistor FLOTOX d'Intel (d'après [42]).

couche de nitrure. Pour abaisser le seuil, une tension négative élevée doit être appliquée sur la grille du transistor.

Les transistors à grille flottante fonctionnent légèrement différemment. Plutôt que d'être emprisonnés dans le nitrure, les électrons sont accumulés sur une grille flottante (en métal ou en polysilicium), isolée électriquement de la grille de commande et du canal du transistor par des couches d'oxyde (SiO_2). Les électrons sont acheminés sur la grille flottante via une région où l'oxyde est très mince. Cette mince couche d'oxyde porte le nom d'oxyde de tunnel. Pour augmenter le seuil du transistor, une tension positive et appliquée sur la grille du transistor alors que les tensions de drain, source et substrat sont reliées à la masse. Les électrons migrent alors du drain vers la grille flottante en passant par l'oxyde de tunnel. Pour abaisser le seuil, le drain est maintenu à une tension positive alors que la grille et le substrat sont reliés à la masse et la source est laissée flottante. La figure 3.10b illustre la vue en coupe d'un transistor FLOTOX (Floating-gate Tunnel-Oxide) de la compagnie Intel.

Un avantage évident des mémoire non-volatiles (MNV) est le fait que même en l'absence de tension d'alimentation, les valeurs des poids peuvent être préservées. Pour les

P153385.PDF [Page: 113 of 191]

autres types de mémoires, il est nécessaire de recharger la matrice synaptique à chaque fois que le système est réalimenté, ou à chaque fois que le temps de rétention est dépassé (rafraîchissement).

La rétention des MNV est de loin supérieure à celle des mémoires à condensateurs, mais elle n'est tout de même pas idéale. En effet, il a été démontré [42] que leur pouvoir de rétention diminuait fortement (et de manière non-linéaire) durant les premières heures suivant l'écriture, pour ensuite se stabiliser. Ceci résulte en une dégradation de la résolution du composant. Si le temps entre les écritures successives est relativement faible, ceci ne pose pas de problèmes. La rétention des transistors à grille flottante est supérieure à celle des transistors à emprisonnement de charges, en raison de l'épaisseur de l'oxyde de tunnel qui, dans le cas des MNV est environ 4 fois plus épaisse (~200Å comparativement à ~50Å pour le MNOS).

Le temps d'écriture des MNV est plutôt élevé. Il s'agit là d'une conséquence directe de la difficulté de charger la grille flottante avec précision. Ainsi, plusieurs impulsions de faible durée doivent être appliquées sur la grille avant d'atteindre la valeur désirée. Par exemple, pour atteindre une résolution de 7 bits, il faut appliquer 180 impulsions de 20µs chacune pour un temps d'écriture total de 3.6ms [39]. Si la vitesse d'écriture n'est pas critique, il est même possible d'atteindre des résolutions allant de 8 à10 bits. Holler *et al.* [39] ajoutent que s'il était possible de charger la grille flottante un seul électron à la fois, un impressionnante résolution de 30 bits pourrait être obtenue.

Finalement, les MNV nécessitent un réseau de commande complexe pour la mise à jour des poids. En fait, ce qui complique le processus, c'est le calcul de la durée des impulsions ainsi que leur fréquence. C'est pourquoi les concepteurs du ETANN d'Intel (nous reviendrons plus tard sur cette puce) ont choisi de déplacer à l'extérieur de la puce les opé-

P153385.PDF [Page: 115 of 191]

rations de mise à jour des poids, simplifiant ainsi grandement la circuiterie interne de la puce.

3.3.2.4 Mémoires analogiques actives

Le dernier type de mémoires analogiques que nous décrivons sont les mémoires dites *actives*. Ces dernières tirent leur nom de la technique de rafraîchissement utilisée pour maintenir la valeur stockée. Ce rafraîchissement peut être soit local, soit distribué. L'avantage d'un rafraîchissement distribué est l'économie de surface résultante pour les éléments de mémoire. Nous allons décrire deux des techniques de rafraîchissement les plus répandues: la boucle A/N-N/A et le rafraîchissement en escalier.

Le principe de la boucle A/N-N/A est illustré à la figure 3.11. L'interrupteur de gauche échantillonne, pendant Φ_1, la tension V_w stockée sur le condensateur C_w. Le convertisseur analogique à numérique transforme alors la représentation analogique en représentation numérique. Ce dernier est conçu de telle sorte qu'une valeur analogique sera arrondie au premier niveau numérique supérieur à cette valeur. De cette façon, le signal sera parfaitement regénéré en autant que sa valeur ne décroisse pas plus qu'un pas de quantification. La figure 3.11 illustre le cas où la tension V_w ne descend pas en dessous du niveau numérique 1010 et est donc regénérée à 1011. Le convertisseur numérique à analogique reconvertit par la suite le signal numérique en signal analogique. L'interrupteur de droite rafraîchit alors la tension de C_w durant Φ_2. Une mise en garde s'impose. Les interrupteurs de la figure 3.11 doivent être conçus de telle sorte que les courants de fuite ne peuvent que *décharger* le condensateur et non le *charger*. Si ce n'est pas le cas, une erreur se produira à chaque rafraîchissement.

Plus le nombre de bits de résolution requis est élevé, plus la fréquence de rafraîchissement devra être élevée. Par contre, pour plusieurs convertisseurs, les temps de conversion

P153385.PDF [Page: 116 of 191]

tendent à s'allonger quand la résolution augmente. La fréquence de rafraîchissement est également déterminée par la vitesse de décharge du condensateur C_w. Si la résolution et la vitesse de décharge sont modérées, il est possible d'utiliser le même circuit de rafraîchissement pour plusieurs poids. Les poids seront alors rafraîchis en alternance. Cette approche permet de sauver beaucoup de surface.

La deuxième technique de mémoires analogiques actives (MAA) que nous décrivons est la technique de rafraîchissement en escalier développée par Vittoz [79]. Le principe est illustré à la figure 3.12. Un signal global de référence V_q est envoyé à toutes les mémoires. Ce dernier est un signal en «escalier» qui comporte autant de niveaux qu'il y a de niveaux de quantification. Chaque circuit de mémoire détermine, en parallèle, si la valeur stockée sur son condensateur C est inférieure à la tension V_q. Si c'est le cas, de la circuiterie additionnelle provoquera une brève impulsion durant laquelle le condensateur C sera rafraîchi avec la valeur de V_q.

3.3.3 Performances

Le problème de la mesure des performances des architectures électroniques analogiques est similaire à celui des architectures optiques: comme il existe très peu de

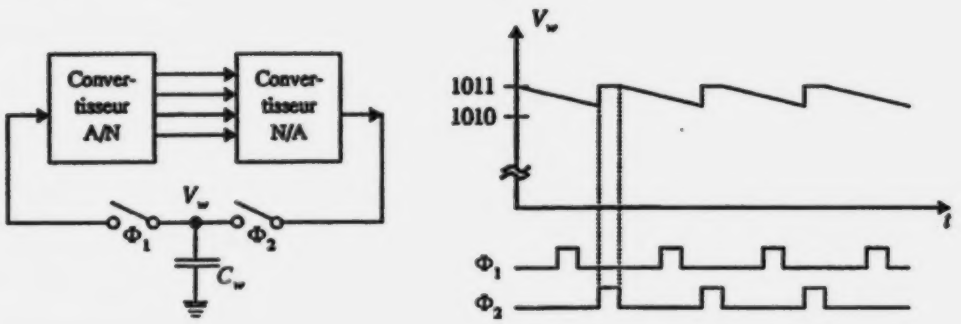


Figure 3.11: Principe de fonctionnement d'une boucle A/N-N/A (d'après [42]).

produits commerciaux (à notre connaissance, deux seulement), les données de performance sont souvent calculées à partir de prototypes expérimentaux. Ainsi, la plupart des architectures n'ont pas bénéficié des nombreuses itérations d'optimisation nécessaires pour une commercialisation rentable. Le lecteur devrait donc interpréter les données repertoriées ci-bas en connaissance de cause.

Commençons tout d'abord par décrire un des rares RNA analogiques commercialement disponibles: le 80170NX de la compagnie Intel. Le 80170NX porte également le surnom de ETANN, pour «Electrically Trainable Analog Neural Network». Cette puce est constituée de 64 neurones analogiques, ainsi que de deux matrices synaptiques de 80x64 synapses chacune, pour un total de 10240 synapses. La première matrice de poids dite «d'entrée» établit les interconnexions entre les 64 tampons d'entrée et les 64 neurones. La seconde matrice, dite «de rétroaction», peut être utilisée dans diverses configurations, dont le mode de rétroaction pour lequel les sorties des neurones sont présentées à l'entrée de cette matrice. Pour chaque matrice de 80x64, une sous-matrice de 16x64 est utilisée pour le biaisage des neurones.

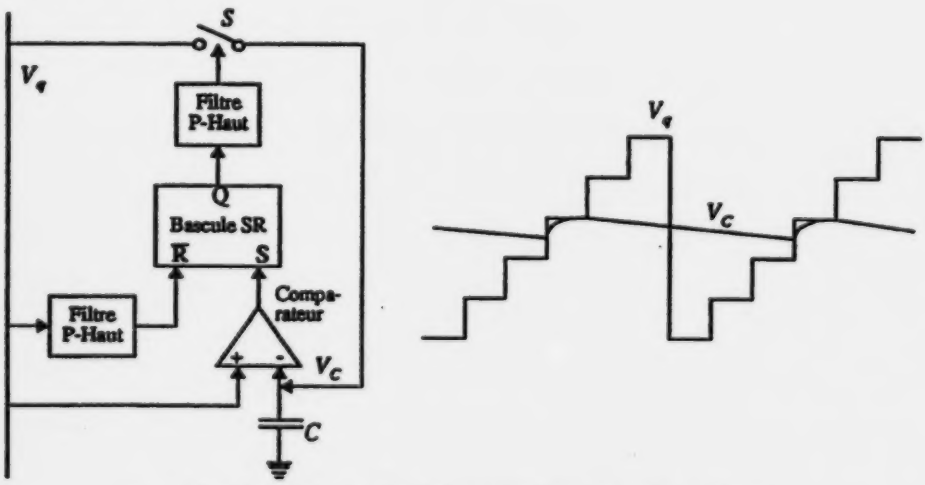


Figure 3.12: Technique de rafraîchissement en escalier (d'après [79]).

P153385.PDF [Page: 118 of 191]

Le temps de traitement se situe entre 1.5 et 3 µs. Si uniquement la matrice d'entrée est utilisée, ceci représente une vitesse entre $64 \times 64 \div 3 \times 10^{-6} \approx 1.4$ GCS et 2.7 GCS (Intel annonce une performance de 2 GCS dans la fiche des spécifications).

La puce utilise la technologie de grille flottante décrite dans la section 3.3.2.3 pour stocker la valeur de ses poids synaptiques. Les poids ne sont adressables qu'un seul à la fois. Combiné au fait que le temps requis pour ajuster la valeur d'un poids est très long, la vitesse d'apprentissage de la puce est donc comparable à celle d'un ordinateur séquentiel de type Von-Neuman. Il existe tout de même un avantage à exécuter l'apprentissage avec la puce plutôt que de simplement charger les poids obtenus par une simulation sur ordinateur. En effet, les imperfections physiques du circuit (faible résolution, non-linéarité) seront ainsi compensées par l'algorithme d'apprentissage.

Notez que la performance d'un système constitué de M puces peut être augmentée par un facteur M. Ainsi, 8 ETANN reliées à un bus d'entrée commun effectueront 8×64 produits scalaires sur un entrant de 64 éléments à une vitesse soutenue d'environ 16 GCS. Ceci suppose que les données peuvent être traitées avec une cadence d'au moins 1 MHz.

Les poids ont une résolution à court terme d'environ 7 à 8 bits et à long terme d'un maximum de 4 bits. La précision des produits scalaires est, quant à elle, de l'ordre de 7 bits.

Boahen *et al.* [8] ont conçu une mémoire associative bidirectionnelle analogique pour laquelle les transistors fonctionnent en mode infraseuil. Il s'agit d'un modèle trois couches du réseau BAM de Kosko [50]. Ce modèle a l'intéressante particularité de ne nécessiter que des poids synaptiques binaires, ce qui simplifie considérablement la conception. Le modèle que les chercheurs ont réalisé comporte 16 neurones d'entrée, 16 neurones de sortie et 7 neurones cachés[1]. Ce réseau est donc capable de stocker un maximum de 7 asso-

P153385.PDF [Page: 119 of 191]

ciations de deux vecteurs de 16 bits. Chaque couche est entièrement interconnectée avec la couche suivante par deux liens. La puce comporte donc un total de $2 \times 2 \times 7 \times 16 = 448$ synapses. Les auteurs ont mesuré une vitesse de rappel d'environ 100 000 patrons à la seconde. Un rappel représente un aller-retour de la couche d'entrée à la couche de sortie, et donc, toutes les synapses effectuent un calcul. Ainsi, la vitesse réelle du réseau est de $100000 \times 448 = 44.8$ MCS.

Dans [33], Graf *et al.* décrivent un réseau constitué de 54 neurones et de 54×54 synapses. Le poids de chaque synapse peut prendre les valeurs -1, 0 ou +1. Elles sont réalisées selon un principe similaire à celui décrit dans la section 3.3.2.1. En général, 4 neurones sont utilisés pour introduire un biais à l'entrant net des autres neurones. Par conséquent, dans nos calculs de performance, nous allons considérer uniquement une matrice de $50 \times 50 = 2500$ synapses. Les auteurs ont mesuré un temps de réponse de l'ordre de la microseconde. Ainsi, la vitesse de cette puce peut atteindre les $2500/1 \times 10^{-6} = 2.5$ GCS. Cependant, avec leur montage expérimental, ce débit de calcul ne pouvait être maintenu en raison de la lenteur avec laquelle le micro-ordinateur chargeait et récupérait les données. En effet, ce dernier nécessitait 25 cycles à 1MHz, pour un délai total de 25 µs, ce qui réduit la vitesse à 100 MCS. Notez que l'article [33] a été publié en 1988, et donc les ressources matérielles dataient de 1987. De nos jours, cette limitation aurait aisément été contournée, car les fréquences d'horloge des microprocesseurs sont souvent supérieures à 25 MHz.

Le dernier circuit que nous décrivons est le fruit d'une collaboration entre Hughes et NOSC (Naval Ocean Systems Center). Il s'agit d'une puce capable d'émuler une couche entière d'un réseau perceptron multicouche. Pour obtenir un réseau de deux couches, par

1. En fait, il y a 14 neurones cachés, 7 neurones pour la propagation directe et 7 pour la rétropropagation. La puce comporte donc un total de 32+14=46 neurones.

P153385.PDF [Page: 120 of 191]

exemple, deux puces peuvent aisément être cascadées. Chaque puce est capable de réaliser une version ternaire de l'algorithme d'apprentissage *Backpropagation*. L'algorithme est dit *ternaire* car les variations des poids sont restreintes à trois types d'incréments constants: négatifs, positifs ou nuls. Les poids sont représentés de manière analogique sur les grilles flottantes d'un multiplieur analogique (la fiche préliminaire ne donne aucune précision quant à la nature de ce multiplieur).

Au total, plus de 100000 synapses et 300 neurones sont disponibles par puce. Le temps de propagation à travers la matrice synaptique et les neurones est de l'ordre de 10 µs, ce qui, pour un réseau de 100000 synapses, se traduit par une vitesse d'environ 10 GCS. En mode d'apprentissage, la vitesse se situe entre 5 et 10 GCS.

Tableau 3.2: Résumé des performances des architectures analogiques décrites dans le texte.

Caractéristiques	Intel 80170NX (ETANN)[a]	BAM de Boahen et al. [8]	AT&T Bell Labs (Graf et al. [33])	Hughes-NOSC
Millions de connexions par seconde (MCPS)	1 400	44.8	2500	10 000
Nombre de neurones	64	46	54	300
Commercialement disponible	✓			✓

a. Les données tabulées sont pour une seule puce ETANN.

Le tableau 3.2 résume les performances des architectures analogiques que nous venons de décrire.

Dans ce chapitre, nous avons étudié les réalisations électroniques de réseaux de neurones artificiels. Celles-ci peuvent être soit numériques, soit analogiques. Le tableau comparatif 3.3 de la page suivante a pour but de résumer les principaux avantages et inconvénients de chaque type d'architecture selon les points de vue pratique et performance. Dans le chapitre suivant, nous confronterons les deux technologies en nous basant sur trois types de critères: les critères de performance, les critères technologiques et les critères d'ordre pratique.

P153385.PDF [Page: 121 of 191]

Tableau 3.3: Résumé des avantages et inconvénients des architectures électroniques numériques et analogiques selon les points de vue pratique et performance.

Architectures	Point de vue	Avantages	Inconvénients
Numériques	Pratique	• Procédés de fabrication éprouvés • Programmabilité • Choix de la topologie d'interconnexion • Compatibilité avec les ordinateurs • Stockage de la mémoire fiable	• Niveau d'intégration (en synapses) faible • Consommation de puissance élevée • Faible tolérance aux pannes
	Performance	• Grands réseaux peuvent être émulés • Précision et exactitude des calculs potentiellement élevées • Possibilité d'apprentissage • Temps de traitement de l'ordre de la nanoseconde • CPS potentiellement élevées	• Haute performance nécessite la mise à l'échelle au niveau système • Nécessite la conversion A/N lorsque les données sont de nature analogique • Organes de calcul doivent être volumineux pour être rapides
Analogiques	Pratique	• Niveau d'intégration élevé • Consommation de puissance potentiellement faible • Tolérance aux pannes • Interface directe avec senseurs analogiques • Mémoires plus compactes	• Difficulté à tester • Peu de puces commercialement disponibles • Difficulté de stockage de mémoire • Mémoires à faible temps de rétention
	Performance	• Hautement parallèles • CPS élevées • Effectue les sommations synaptiques instantannément par la loi de Kirchoff	• Apprentissage doit souvent être externe • Précision et exactitude des calculs faibles • Temps de traitement de l'ordre de la microseconde

CHAPITRE IV COMPARAISON OPTIQUE/ ÉLECTRONIQUE

Tout au long des chapitres II et III, nous avons eu l'occasion de calculer les performances ainsi que d'apprécier certaines considérations d'ordre pratique pour les technologies optique et électronique. Une emphase particulière a été mise sur la performance. Cependant, le choix d'une technologie ne se limite pas seulement à sélectionner celle qui calculera le plus de connexions à la seconde. Chaque application ayant ses besoins particuliers, une comparaison moins superficielle s'impose. Dans ce chapitre, nous allons confronter les deux technologies en se basant sur plusieurs autres critères de comparaison. À notre avis, les plus importants critères sont:

- Vitesse;
- Capacité d'interconnexion;
- Précision et exactitude dans les calculs;
- Fiabilité et tolérance aux pannes;
- Intégration;
- Mise à l'échelle (scalability);
- Flexibilité;
- Consommation de puissance;
- Poids et dimensions physiques;
- Coût économique et maturité.

P153385.PDF [Page: 123 of 191]

Il est important, lors du choix d'une technologie pour une application donnée, de bien comprendre les forces et les faiblesses de chacune d'elles. Évidemment, la supériorité d'une technologie ne se manifeste pas sur tous les critères, c'est pourquoi les besoins de l'application devront préalablement être pondérés avant d'arrêter notre choix sur une technologie.

Il est également important de reconnaître à quel point les critères énumérés ci-dessus sont interdépendants. Par exemple, plus la vitesse d'un circuit est élevée, plus sa consommation de puissance est élevée; plus un système exploite une technologie d'avant-garde, plus son coût augmente en conséquence; plus un système est flexible, plus son niveau d'intégration diminue, et ainsi de suite. Tenter d'optimiser tous les critères en même temps, c'est tenter d'obtenir la quadrature d'un cercle: c'est tout simplement impossible.

Dans ce qui suit, nous décrivons chacun des critères et nous confrontons les technologies optique et électronique. Nous avons choisi de répartir ces critères en trois grandes classes:

- *les critères de performance* qui regroupent tous les critères reliés à la performance du système: la vitesse, la capacité d'interconnexion, la précision et l'exactitude des calculs;
- *les critères technologiques* qui dépendent fortement d'une technologie donnée comme la fiabilité et la tolérance aux pannes, l'intégration, la mise à l'échelle et la flexibilité;
- *les critères d'ordre pratique* comme la consommation de puissance, le poids et les dimensions physiques, le coût économique et la maturité.

4.1 Critères de performance

4.1.1 Vitesse

Les chapitres II et III ont déjà traité en détail des vitesses des diverses architectures optique et électronique de réseaux de neurones. Dans cette section, nous nous contente-

P153385.PDF [Page: 124 of 191]

rons de résumer ces données afin de tirer des conclusions générales. Le tableau 4.1 résume donc les résultats de ces chapitres.

Tableau 4.1: Résumé des vitesses des architectures optiques et électroniques. Toutes les données sont en connexions/seconde.

Architectures		Variation	Moyenne	Déviation standard
Optiques	À processeurs algébriques optiques (TVCL)	$[1.92\times10^{6}, 262.14\times10^{6}]$	59×10^{6}	100×10^{6}
	À processeurs algébriques optiques (acousto-optique)	$[620\times10^{6}, 249\times10^{9}]$	126×10^{9}	142×10^{9}
	À corrélateurs optiques	$[128\times10^{9}, 10.5\times10^{15}]$	3143×10^{12}	5000×10^{12}
Électroniques	Numériques	$[450\times10^{6}, 10\times10^{9}]$	1.1×10^{9}	3×10^{9}
	Analogiques	$[44.8\times10^{6}, 10\times10^{9}]$	3.5×10^{9}	4.5×10^{9}

La vitesse d'une architecture neuronale est exprimée en connexions par seconde. Ainsi, il existe deux façons de maximiser la vitesse: minimiser le temps de traitement pour une connexion ou maximiser le nombre de connexions que l'on calcule en parallèle. Les architectures optiques sont très parallèles (voir section 4.1.2). Donc, même si le temps requis pour exécuter une connexion est relativement élevé (de l'ordre de la milliseconde), ces architectures arrivent tout de même à atteindre des performances impressionnantes. Le tableau nous montre également que les architectures à corrélateurs optiques sont de loin les plus rapides, suivies des architectures à processeurs algébriques optiques utilisant des matrices acousto-optiques. Celles utilisant des matrices TVCL sont en dernière place, et elles sont même surclassées par l'électronique.

Les architectures électroniques analogiques et numériques ont des performances sensiblement comparables. Pour l'analogique, la vitesse de calcul pour une interconnexion est généralement plus rapide que pour les architectures optiques, mais le parallélisme est moindre. Pour le numérique, la vitesse de calcul pour une interconnexion est souvent très

P153385.PDF [Page: 125 of 191]

rapide (souvent de l'ordre de quelques nanosecondes), mais comme nous l'avons vu dans la section 3.2.1, en raison du multiplexage des ressources de calcul et de communication, le parallélisme est moindre que pour les réalisations analogiques.

4.1.2 Capacité d'interconnexion

La capacité d'interconnexion est définie comme l'aisance avec laquelle une technologie permet d'interconnecter plusieurs processeurs. À cet égard, il n'est pas surprenant d'entendre que la technologie optique est souvent référée comme étant une *technologie d'interconnexion*. Cette supériorité est attribuable à deux facteurs. Tout d'abord, une propriété intéressante de la lumière est qu'elle n'interfère pas. Deux faisceaux lumineux peuvent se croiser sans interférence. Cette propriété permet d'obtenir une densité d'interconnexion surprenante. Un autre facteur important est que la lumière ne nécessite pas nécessairement un milieu dédié pour se propager, comme c'est le cas pour l'électronique. Un contre-exemple est la fibre optique, mais cette technique n'est intéressante que lorsque les distances à parcourir deviennent très grandes. Pour les applications de réseaux de neurones où les distances sont généralement très courtes, il est préférable de laisser les faisceaux lumineux se propager dans l'air libre, ce qui, de paire avec la propriété de non-interférence, contribue également à accroître la densité d'interconnexion. À titre d'exemple, la lentille d'une caméra 35mm, qui est un cas particulier d'un système d'interconnexion optique, peut traiter jusqu'à un million de connexions optiques indépendantes [24].

À l'inverse, l'électronique ne jouit d'aucune des deux propriétés décrites plus haut. Un signal électrique étant la propagation de particules chargées, la proximité d'un autre signal provoquera évidemment un certain niveau d'interférence. De plus, à basses fréquences, ces signaux doivent absolument se propager dans un milieu conducteur: le fil électrique. Pour des applications où le degré d'interconnexion est très élevé, comme c'est le cas pour les réseaux de neurones, il n'est pas surprenant de constater que le métal utilisé pour les

P153385.PDF [Page: 126 of 191]

fils représente souvent un proportion considérable de la surface totale d'un circuit intégré, car n'oublions pas que la topologie d'interconnexion doit obligatoirement être bidimensionnelle, en raison de la nature planaire d'un circuit intégré.

Il est clair, d'après notre discussion sur les réalisations optiques au chapitre II, que certaines architectures profitent davantage de la puissance d'interconnexion de la technologie. Les architectures à corrélateurs optiques, conçues pour traiter des images bidimensionnelles, en sont un exemple.

4.1.3 Précision et exactitude dans les calculs

Avant d'entreprendre une discussion sur la précision et l'exactitude des calculs dans les architectures matérielles, il importe tout d'abord de bien définir ces deux termes. La *précision* d'un nombre est une mesure de la plus petite variation de ce nombre que l'on puisse représenter. L'*exactitude* d'un nombre est une mesure de la plus petite variation de ce nombre que l'on puisse distinguer avec certitude. Par exemple, un nombre peut avoir une précision de 8 bits mais une exactitude de seulement 3 bits. On dira alors que le nombre possède 3 bits significatifs. Nous pouvons aussi utiliser la définition de l'exactitude pour mesurer l'incertitude multiplicative d'un nombre. Par exemple, si nous voulons représenter la valeur v par le nombre u, mais que l'incertitude multiplicative sur v est e, alors nous pouvons affirmer avec certitude que

$$(1-e)v \leq u \leq (1+e)v. \tag{4.1}$$

On dira alors que l'exactitude de u sera de k bits où k est le plus grand nombre entier qui satisfasse[1]

$$k \leq 1 + \log_2 e^{-1}. \tag{4.2}$$

1. Voir [38]

P153385.PDF [Page: 127 of 191]

Dans les architectures analogiques, les nombres sont représentés par des valeurs physiques, comme des courants ou des tensions. Ces valeurs sont continues et donc théoriquement, possèdent une précision infinie. Cependant, leur exactitude est limitée par la présence de bruits thermique et quantique. Ces bruits ne sont cependant pas les seuls facteurs qui contribuent à limiter l'exactitude des nombres analogiques. Un autre facteur important provient des variations locales des paramètres des transistors, variations qui sont imprévisibles. De plus, nous avons vu dans la section 1.3.2 que le transistor peut être utilisé comme multiplieur. Or, le transistor ne se comporte comme un multiplieur que dans une petite région de sa courbe d'opération, i.e. pour des valeurs de V_{ds} faibles. L'exactitude du courant à la sortie du multiplieur sera donc déterminée par deux facteurs: le produit des incertitudes des deux opérandes et la distorsion non-linéaire introduite par le transistor qui ne peut qu'*approximer* la multiplication. Nous ne pouvons pas échapper au premier facteur: l'exactitude du produit dépendra toujours de l'exactitude des opérandes. Quant au deuxième facteur, nous avons deux choix, soit le tolérer, soit concevoir des organes de calcul qui imitent mieux l'opération algébrique désirée (dans notre exemple, la multiplication). Dans certaines applications, cette perte d'exactitude peut être tolérée. Par exemple, dans [53], les auteurs utilisent un transistor pour réaliser le produit synaptique dans une architecture compétitive. Pour une telle architecture, il suffit que la déviation du comportement linéaire soit monotone croissante. Ainsi, l'ordre relatif des entrants nets (i.e. les produits scalaires) sera préservé (i.e. du plus grand au plus petit) et le comportement du réseau ne sera pas affecté par cette perte d'exactitude.

De façon générale, il est très difficile de prévoir si les pertes d'exactitude provoquées par les organes de calcul matériels auront un impact sur la performance du réseau. Ainsi, lorsque l'application n'est pas connue à l'avance, il est préférable d'opter pour le deuxième choix qui est de concevoir des organes qui approximent mieux l'opération algé-

P153385.PDF [Page: 128 of 191]

brique désirée. Cependant, les meilleurs multiplieurs analogiques sont incapables de franchir la barrière des 8 bits d'exactitude, la moyenne se situant autour de 6 bits [38].

L'avantage du numérique devient alors évident. La précision des nombres représentés dans une notation numérique peut être arbitrairement augmentée. Bien entendu, il faut s'assurer que l'application puisse supporter une exactitude comparable à la précision pour éviter le gaspillage. En effet, les entrées d'un système sont souvent analogiques et leur traduction dans la représentation numérique s'effectue avec une exactitude et une précision limitées. Un autre avantage marqué du numérique vis-à-vis l'analogique est que les organes de calculs peuvent, si on le désire, être conçus pour ne pas introduire de perte d'exactitude due au matériel. Bien entendu, tout comme en analogique, l'exactitude à la sortie de l'organe de calcul dépendera quand même de l'exactitude des opérandes. De plus, en numérique, même si les erreurs dues au bruit sont présentes, elles sont tellement rares comparativement à l'analogique qu'elles sont presque toujours négligées. La raison est qu'en numérique, les nombres sont représentés en base 2. Chaque bit d'information dispose de la plage dynamique maximale disponible: de 0 volts à V_{DD} volts. Cette plage dynamique est divisée en trois régions: la région du "1" valide, celle du "0" valide et une région de garde entre ces deux régions. Typiquement, la région de garde représente de 20% à 30% de la plage dynamique, ce qui laisse jusqu'à 40% pour chaque niveau logique. Cette marge est amplement suffisante pour absorber la presque totalité des erreurs ou perturbations. Dans un circuit analogique, la plage dynamique 0-V_{DD} doit être partagée entre tous les niveaux que peut prendre la valeur analogique. De plus, certains circuits n'opèrent convenablement que dans une fraction de la plage 0-V_{DD}. C'est le cas, par exemple, de l'amplificateur de transconductance utilisé comme multiplieur (voir 1.3.3).

À ce jour, les besoins en exactitude des nombreuses architectures neuronales ne sont pas encore bien spécifiés. Certains résultats existent déjà pour le réseau *backpropagation*:

P153385.PDF [Page: 129 of 191]

en phase d'apprentissage, l'exactitude doit être supérieure à 6 bits [36] pour pouvoir échapper aux minimums locaux dans la surface d'erreur globale. De façon plus générale, il a été remarqué que la plupart des réseaux nécessitent plus d'exactitude en phase d'apprentissage qu'en mode de traitement normal. Ainsi, il est de notre avis que si l'application n'est pas connue à l'avance, l'approche numérique est très fortement préférable.

Le niveau d'exactitude qu'il est possible d'obtenir avec la technologie optique est comparable à celui qu'il est possible d'obtenir en électronique analogique. Notons cependant qu'il existe une façon d'augmenter l'exactitude des calculs en optique en adoptant diverses stratégies mais la vitesse en est sévèrement réduite de telle sorte que l'utilisation de l'optique n'est plus justifiée (voir page 35). Sans stratégie d'augmentation d'exactitude, l'exactitude maximale que peut livrer une architecture optique est de l'ordre de 8 bits. Cette exactitude varie selon le type d'architecture. Par exemple, les architectures à interconnexions holographiques procurent en général une précision moindre que les architectures à processeurs algébriques optiques.

Les architectures électroniques analogiques réussissent à obtenir des précisions dans leurs poids variant de quelques bits (3 ou 4) pour les mémoires numériques contrôlant une valeur analogique quantifiée, jusqu'à un peu moins de 12 bits pour les mémoires non-volatiles. Cependant, pour supporter de telles précisions, les multiplieurs analogiques devront être relativement complexes et donc occuper une grande surface. N'oublions pas que leur exactitude sera quand même restreinte à un maximum de 8 bits.

Il est important de montrer le lien étroit qui existe entre exactitude et vitesse. En effet, l'exactitude croit de façon inversement proportionnelle avec la vitesse. Tout module, qu'il soit analogique ou numérique, doit transmettre le résultat de son calcul à une *charge*. Cette charge peut être l'entrée capacitive d'une porte logique CMOS pour un circuit numérique,

P153385.PDF [Page: 130 of 191]

ou l'entrée capacitive d'un amplificateur différentiel pour un circuit analogique, par exemple. Dans les deux cas, une constante de temps est associée aux variations de tension aux bornes de cette charge. Cette constante de temps est proportionnelle au produit de la résistance de sortie du module et de la capacité d'entrée du module suivant agissant comme charge (i.e. $\tau = RC$). La vitesse de charge du condensateur est donc une constante, pour une puissance donnée. Si on accélère la vitesse du circuit (pour un circuit numérique, on augmente la fréquence d'horloge alors que pour un circuit analogique, on considère la sortie comme étant valide plus tôt), le condensateur n'aura pas le temps de se charger à la valeur de tension voulue. Les marges d'erreur pour les circuits numériques vont alors se rétrécir et cela peut aller jusqu'à causer des erreurs dans le traitement. Le principe est le même pour les circuits analogiques. Cependant, la dégradation de l'exactitude avec l'augmentation de la vitesse est plus graduelle.

4.2 Critères technologiques

4.2.1 Fiabilité et tolérance aux pannes

Une caractéristique intéressante d'un réseau de neurones est sa tolérance aux pannes. Cette propriété est une conséquence du parallélisme des interconnexions entre les neurones. En effet, si quelques connexions sont coupées ou que leur poids est gelé à une valeur fixe, le comportement global du réseau ne sera presque pas affecté. Voyons à présent si cette propriété est préservée lorsque le réseau est réalisé en matériel.

Nous avons vu dans la section 3.3.1 que les architectures analogiques utilisent généralement le parallélisme des interconnexions, c'est-à-dire qu'il existe un circuit synaptique pour chaque interconnexion. Un défaut provoquant une panne dans un des circuits synaptiques ne sera en général pas catastrophique. Cependant, si un défaut se produit sur la ligne qui relie tous les courants synaptiques, le dommage est beaucoup plus considérable.

P153385.PDF [Page: 131 of 191]

Pour les architectures numériques où, bien souvent, les ressources de calcul et de communication sont multiplexées dans le temps, les pannes sont beaucoup plus dévastatrices. Par conséquent, des techniques artificielles de tolérance aux pannes doivent être prévues pour permettre un rendement satisfaisant. Une exemple d'une telle technique est la duplication des bus de communication.

Les architectures optiques sont probablement les plus «parallèles», mais elles ne sont pas nécessairement les plus tolérantes aux pannes. En effet, les «pannes» en optique peuvent se manifester de deux façons, toutes deux étant la conséquence de défauts physiques. Le premier type de pannes, comparable aux pannes électroniques, est dû à la présence de poussières ou d'égratignures sur les lentilles, les hologrammes ou les modulateurs spatiaux de lumière. Ces pannes peuvent généralement être absorbées par le parallélisme de l'architecture optique. Le deuxième type de pannes, beaucoup plus grave cette fois, est dû aux erreurs d'alignement des composants. En effet, un montage optique nécessite un alignement très précis de ses composants (souvent de l'ordre du dixième de millimètre). Ces pannes peuvent survenir à la fabrication (i.e. sur la chaîne de montage) ou durant l'utilisation (suite à un choc physique). Ce besoin de précision dans l'alignement des composants confère aux architectures optiques une certaine fragilité mécanique et une forte sensibilité aux vibrations.

4.2.2 Intégration

Règle générale, les architectures analogiques atteignent des niveaux d'intégration supérieurs à ceux des architectures numériques. La raison est que les constituants des architectures analogiques sont relativement simples en compacts. Pour les architectures électroniques analogiques, ce qui détermine le niveau d'intégration, c'est le design du circuit synaptique qui, généralement, comprend le circuit pour mémoriser le poids. Bien que la surface de ce circuit croisse avec la complexité du design selon le niveau de précision

P153385.PDF [Page: 132 of 191]

requis, le niveau d'intégration est néanmoins supérieur à celui des architectures électroniques numériques. En fait, la dimension des synapses numériques est tellement grande comparativement à celle des synapses analogiques qu'il est impossible d'intégrer suffisamment de neurones par puce pour des applications réelles. C'est pourquoi bon nombre de concepteurs de RNA numériques ont préféré multiplexer les interconnexions pour ainsi allouer plus d'espace à la partie «traitement» du RNA.

Il ne faut pas oublier de mentionner que les concepteurs de circuits numériques ont souvent accès à des procédés de fabrication généralement beaucoup plus éprouvés que les procédés analogiques. Le marché d'électronique de masse, en raison de la popularité des microprocesseurs, est littéralement dominé par le numérique. En conséquent, des efforts considérables sont déployés par les manufacturiers de circuits intégrés pour constamment améliorer leurs procédés de fabrication. En 1994, les procédés CMOS à 0.8 micromètres sont devenus courants. C'est pourquoi, même si du point de vue architectural, les RNA analogiques ont un potentiel d'intégration supérieur aux RNA numériques, ces derniers regagnent un peu de terrain en profitant des procédés de fabrication numérique de pointe.

Dans son sens large, intégration est avant tout synonyme de *densité*. Pour les circuits électroniques, cette densité (planaire) est obtenue sur un substrat de semi-conducteur. À l'inverse, la fonctionnalité des architectures optiques n'est pas contenue uniquement sur un substrat planaire, mais est plutôt répartie dans un volume. En effet, lentilles, hologrammes et MSL se partagent tous le même *volume de traitement*. Donc, lorsqu'il est question d'intégration en optique, il est sous entendu qu'il est question de densité *volumique* et non *planaire*, comme pour les architectures électroniques.

Comme très peux de RNA optiques commerciaux existent, il est difficile de prévoir le niveau d'intégration de ces derniers. La plupart des montages expérimentaux utilisent des

P153385.PDF [Page: 133 of 191]

composants encombrants. Il est certainement possible d'optimiser un montage pour minimiser ses dimensions physiques, mais il existe peu de littérature sur le sujet. Il nous est tout de même possible de tirer certaines conclusions. Prenons l'architecture à corrélateurs optiques, par exemple. Son impressionnante capacité de calcul est tout simplement impossible à obtenir avec une seule puce ITGE; elle ne peut qu'être approchée au niveau système par une mise à l'échelle (voir section 4.2.3 ci-bas). Donc, il n'est pas exclu que les dimensions d'une carte ou d'une série de cartes électroniques soient comparables à celles du montage optique optimisé en dimensions.

Il est également possible en optique d'intégrer en deux dimensions, comme en électronique. L'intégration 2-D optique n'est définitivement pas aussi avancée que l'intégration électronique. Il existe quelques exemples d'architectures optiques intégrées mais leurs performances sont souvent beaucoup moindres que celles de la version non-intégrée de la même architecture. La raison est simple: ce qui fait la force de la technologie optique, c'est son pouvoir d'interconnexion (voir 4.1.2). Ce dernier est obtenu par l'exploitation de la troisième dimension pour la propagation des signaux lumineux. En optique intégrée, cette troisième dimension est à toutes fins pratiques perdue. Plus précisément, dans un circuit optique intégré, la lumière se propage dans un milieu diélectrique délimité par deux autres milieux diélectriques d'indice de réfraction moindre. Par le principe de réflexion totale interne utilisé dans les fibres optiques, le faisceau lumineux est confiné dans le milieu à indice de réfraction élevé.

4.2.3 Mise à l'échelle

La mise à l'échelle idéale est définie comme la capacité d'augmenter le nombre de processeurs d'une architecture parallèle (et donc traiter plus de données) tout en maintenant le temps de traitement des processeurs constant. Ainsi, le temps de réponse d'un réseau de N neurones sera constant, peu importe la valeur de N.

Mettre à l'échelle un RNA électronique ne signifie pas simplement augmenter la surface de silicium du circuit. Il existe une limite à la taille des circuits électroniques intégrés que l'on puisse fabriquer. Outre le coût, la raison fondamentale qui limite la surface d'un circuit est le *rendement*. Par conséquent, la mise à l'échelle des architectures électroniques sera toujours obtenue au niveau système en cascadant plusieurs puces ensemble. Une technique de plus en plus à la mode consiste à déposer plusieurs circuits sur un même substrat pour ensuite les interconnecter à l'aide de traces à même le substrat [81]. Cette technique porte le nom d'*intégration sur plaquette* (Wafer Scale Integration). Une autre technique, moins coûteuse cette fois, consiste à placer les puces sur un circuit imprimé. C'est la technique adoptée par plusieurs manufacturiers de cartes accélératrices. Cependant, les délais de propagation sont importants lorsqu'un signal doit quitter un circuit intégré en raison de la grande capacité parasite des plots de sortie. C'est pourquoi la technique d'intégration sur plaquette est plus performante. Cette dernière permet également des économies substantielles en éliminant la nécessité d'emboîter chaque puce, les boîtiers représentant une importante partie du coût total d'un circuit intégré.

En raison des délais dus aux interconnexions, il est évident que la mise à l'échelle idéale d'une architecture neuronale électronique est impossible en raison de sa forte connectivité. Cela ne veut pas dire qu'il est impossible de simuler beaucoup de neurones, mais plutôt que le temps requis pour calculer un grand nombre de neurones sera supérieur à celui requis pour un plus petit nombre de neurones. Un type de réseaux échappe à cette règle: les réseaux à interconnexions locales. Pour ce type de réseaux, les délais imputables aux interconnexions ne sont pas fonction des dimensions du circuit puisque chaque processeur communique uniquement avec les processeurs qui lui sont voisins. Le modèle de la rétine de l'œil humain développé par Mead et ses collègues en est un exemple [56].

P153385.PDF [Page: 135 of 191]

Le problème de la mise à l'échelle des RNA optiques est sensiblement différent de celui des RNA électroniques. Étant donné la nature parallèle du traitement optique, la mise à l'échelle idéale est en théorie toujours possible. En pratique, elle est malheureusement limitée par la taille des composants optiques tels les lentilles, les hologrammes et les modulateurs spatiaux de lumière. En effet, le nombre de canaux qu'un système optique peut traiter en parallèle est donné par le produit de la plus petite surface de traitement du système optique (perpendiculaire à l'axe optique) par sa résolution spatiale, c'est-à-dire la fréquence spatiale maximale qu'il puisse distinguer, donc traiter. La plus petite surface de traitement dans un système porte le nom d'*ouverture numérique* et son produit avec la résolution spatiale porte le nom de *produit espace-bande*. Donc, pour mettre un système optique à l'échelle, il faut soit augmenter son ouverture numérique, soit augmenter sa résolution spatiale. La première option n'est pas souvent possible en raison des effets secondaires qui en résulte. Par exemple, une façon d'augmenter l'ouverture numérique d'un système optique est d'augmenter le diamètre des lentilles. Or, plus les lentilles sont grosses, plus les problèmes d'aberration sont susceptibles de se manifester. L'autre solution est d'augmenter la résolution du système. Souvent, cette dernière est imposée par les modulateurs spatiaux de lumière comme les TVCL et les valves optiques. Nous avons déjà vu que la résolution des TVCL est limitée par l'espace inter-pixel requis pour l'adressage des pixels (<50 pixels/mm). La marge de manœuvre est donc réduite pour ce type de MSL. Les valves optiques, en revanche, ne souffrent pas de ce problème. (~100 pl/mm)[1]

4.2.4 Flexibilité

Il existe deux critères généralement acceptés pour juger de la flexibilité d'une réalisation neuronale. Le premier critère important est la possibilité de modifier le poids des

1. La résolution des MSL discrets s'exprime en pixels par mm. Pour un MSL continu, la résolution s'enprime en paires de lignes affichables par mm. Les deux mesures sont cependant équivalentes.

P153385.PDF [Page: 136 of 191]

interconnexions de façon adaptative. À ce jour, peu de réalisations, quelle que soit la technologie de réalisation, peuvent se vanter de remplir cette exigence. Bien souvent, la phase d'apprentissage doit se faire à l'extérieur du système. Les poids sont par la suite injectés dans le système et ne peuvent plus être modifiés. Il est de l'avis de beaucoup de chercheurs que l'apprentissage adaptatif est à la base de l'avantage inhérent des architectures neuronales vis-à-vis les algorithmes traditionnels en intelligence artificielle. Une architecture capable d'apprentissage adaptatif a donc une longueur d'avance sur les autres architectures.

La complexité des calculs requis pour l'apprentissage adaptatif est souvent le facteur prohibitif. En effet, certaines règles d'apprentissage comme le *backpropagation* requièrent le calcul d'une dérivée, de l'élévation au carré, etc. Souvent, ces calculs doivent être exécutés avec un niveau d'exactitude suffisamment élevé pour permettre au réseau d'atteindre la convergence. C'est pourquoi on retrouve davantage de réalisations électroniques numériques avec apprentissage que de réalisations analogiques ou optiques.

Le deuxième critère qui permet de juger de la flexibilité d'une architecture est le nombre d'algorithmes neuronaux différents qu'il est possible d'émuler sur cette architecture. Un critère qui découle directement de cette exigence est la possibilité de reconfigurer les interconnexions entre les neurones.

Une «règle du pouce» en ce qui concerne la flexibilité des architectures neuronales pourrait être définie comme tel: plus une réalisation mise sur les propriétés physiques des composants qui la constitue, plus la flexibilité du système résultant diminue. Dans les architectures électroniques numériques, les composants principaux (les transistors) sont exploités dans leur mode d'utilisation le plus simple: le mode *interrupteur*. La commande de ces interrupteurs se fait à l'aide de seulement deux niveaux logiques (haut et bas). La

P153385.PDF [Page: 137 of 191]

fonctionnalité d'un réseau d'interrupteurs peut donc être aisément programmée par un mot binaire, d'où la grande flexibilité de ces architectures.

La situation est différente pour les architectures électroniques analogiques. Pour ces dernières, les propriétés physiques du transistor sont davantage exploitées. Bien souvent, le transistor est utilisé comme amplificateur, plutôt que comme interrupteur. L'amplificateur de transconductance est un exemple de circuit analogique fréquemment utilisé [56]. Ce genre de circuit possède une plage d'opération non-linéaire qui n'est pas nécessairement appropriée pour toutes les applications. De plus, la précision réduite des calculs analogiques limite également l'applicabilité de l'analogique.

En général, les architectures analogiques qui procurent un certain degré de flexibilité font appel à un moment ou un autre à l'électronique numérique. Les mémoires numériques contrôlant une valeur analogique quantifiée sont un exemple (voir section 3.3.2.1 en page 86). Un autre exemple est la puce ETANN d'Intel. Cette dernière requiert l'assistance d'un contrôleur numérique complexe (externe) pour ajuster la valeur de ses poids internes.

Tout comme les architectures électroniques analogiques, les architectures optiques sont peu flexibles. Le meilleur exemple de ce manque de flexibilité sont les architectures à interconnexions holographiques. Ces dernières doivent prendre bien soin d'être compatibles avec les propriétés physiques du cristal photoréfractif. Autrement dit, leur dynamique est indissociable des propriétés physiques du cristal. Il n'est pas surprenant que les architectures optiques qui sont le moindrement flexibles le sont grâce à l'électronique. Les architectures à processeurs algébriques optiques, par exemple, utilisent des téléviseurs à cristaux liquides pour moduler la lumière incidente et agir ainsi à titre de matrice synapti-

P153385.PDF [Page: 138 of 191]

que. Ces derniers requièrent l'assistance de l'électronique pour contrôler les niveaux de transmittance des pixels.

4.3 Critères d'ordre pratique

4.3.1 Consommation de puissance

La consommation de puissance est un critère de design important pour les architectures de réseaux de neurones. Elle l'est plus particulièrement pour les architectures électroniques puisque la puissance consommée doit être entièrement dissipée en chaleur. Ceci résulte en une forte augmentation de la température de la surface du circuit intégré (il n'est pas rare de mesurer des températures allant jusqu'à 70 degrés celcius). Malheureusement, les propriétés d'un circuit électronique sont sévèrement dégradées lorsque sa température est trop élevée. Il est donc important de tenter de minimiser la consommation de puissance d'un circuit.

Dans un circuit électronique, qu'il soit analogique ou numérique, on distingue deux types de puissance, la puissance *statique* et la puissance *dynamique*. La puissance statique est la puissance qui est consommée par un circuit lorsque ce dernier est en état de repos. La puissance dynamique, quant à elle, est la puissance dissipée uniquement lors des changements d'état du circuit. Un exemple de puissance statique est celle requise pour réaliser la polarisation des divers constituants d'un circuit analogique.

La consommation de puissance d'un circuit CMOS numérique est relativement simple à estimer, puisqu'elle est directement reliée aux changements d'état dans le circuit. Il est facile de constater que plus la fréquence d'opération d'un circuit CMOS est élevée, plus la consommation de puissance sera élevée. En effet, la puissance moyenne est définie comme le rapport de l'énergie consommée sur le temps requis pour consommer cette énergie. En supposant une vitesse de transisition constante (se traduisant par une puissance de

P153385.PDF [Page: 139 of 191]

court-circuit constante), comme l'énergie requise pour charger la capacité d'entrée d'une porte logique est constante, la puissance moyenne augmente donc si on diminue le temps requis pour effectuer cette transistion. Ce phénomène fait la force des circuits CMOS. En effet, dans la plupart des cas, seule une fraction des portes logiques dans le circuit changeront d'état et dissiperont ainsi de la puissance.

Dans un circuit analogique, même les sous-circuits qui ne changent pas d'état consomment de la puissance de polarisation. Cependant, alors que pour les circuits numériques, un changement d'état est caractérisé par une variation de 0V à 5V, pour les circuits analogiques, les changements d'état ne sont pas nécessairement caractérisés par des grandes variations de signal. Illustrons ceci par un exemple simple. Supposons que la sortie d'un multiplieur soit reliée à un condensateur de capacité C. Supposons que cette sortie doive passer de V_1 à V_2. L'énergie requise pour cette transition est

$$\Delta E = \frac{1}{2}C(V_2^2 - V_1^2). \tag{4.3}$$

Pour un multiplieur numérique, la sortie est quantifiée, de sorte que passer de V_1 à V_2 se traduit par le changement d'état de plusieurs bits à la sortie du multiplieur. Pour chaque bit qui change d'état, l'énergie requise sera $\frac{1}{2}C_g V_{DD}^2$, où C_g est la capacité à la sortie de chaque bit du registre. On voit clairement que l'architecture analogique économise l'énergie en limitant les variations du signal de sortie. Une comparaison plus poussée devrait évidemment tenir compte de la puissance de polarisation pour le circuit analogique et de la puissance dissipée près du seuil de la porte lors d'une transistion pour un circuit numérique CMOS.

Bien qu'en première approximation, les circuits analogiques semblent consommer moins de puissance, n'oublions pas qu'en général, leur niveau d'intégration, en nombre de synapses ou neurones par puce, est supérieur à celui des circuits numériques. Par consé-

P153385.PDF [Page: 140 of 191]

quent, la puissance dissipée par unité de surface pourrait être sensiblement la même. C'est pourquoi, de plus en plus, les concepteurs de circuits analogiques se tournent vers un fonctionnement en mode infraseuil pour diminuer considérablement la puissance. Un transistor opéré en mode infraseuil génère des courants de très faible amplitude (voir section 1.3). Comme la puissance moyenne est le produit du courant et de la tension d'alimentation, la puissance totale du circuit est donc réduite. Cependant, rien n'est gratuit: plus les courants sont de faible amplitude, plus ils risquent de se noyer dans le bruit et plus les délais augmentent.

Afin de comparer l'électronique avec l'optique, référons nous aux travaux de J.W. Goodman [68]. Ce dernier a comparé les consommations de puissance requises pour réaliser deux types d'interconnexions: intra-puce (longueur d'interconnexion de l'ordre du mm) et inter-puces (longueur de l'ordre du cm). Un exemple numérique illustre qu'un lien intra-puce typique (entre deux portes logiques) nécessite une puissance d'environ 35µW pour un lien électrique et une puissance d'environ 1mW pour un lien optique. Il est donc évident que dans un tel cas, le lien électrique est beaucoup plus avantageux que le lien optique. Notez que dans le cas du lien optique, la majeure partie de la puissance est utilisée pour polariser la source laser. La puissance requise pour charger le detecteur à l'autre bout du lien devient alors négligeable.

Pour un lien inter-puce, un exemple numérique similaire démontre que la puissance requise pour un lien électronique est de 175µW, alors que celle requise pour un lien optique est encore de 1mW. L'avantage du lien électronique n'est pas aussi évident que dans l'exemple précédent, bien que la puissance requise par le lien électronique soit tout de même inférieure de près d'un ordre de grandeur à celle pour le lien optique. Les deux exemples démontrent que l'utilisation de liens optiques uniquement sur la base de la puissance n'est pas justifiable. Cependant, le fait que les interconnexions optiques soient

P153385.PDF [Page: 141 of 191]

exemptes du problème d'interférence mutuelle est un sérieux avantage en faveur de leur utilisation.

Mentionnons pour terminer que si un laser est utilisé pour réaliser plusieurs liens, des économies de puissance par lien peuvent être réalisées puisque la puissance de polarisation sera en quelque sorte «amortie» sur plusieurs liens.

4.3.2 Poids et dimensions physiques

Les dimensions physiques des systèmes optiques sont, en général, plus grandes que celles des systèmes électroniques. Ces dimensions sont non seulement déterminées par les dimensions des composants (lentilles, MSL, hologrammes), mais aussi par les distances entre ces composants, qui dépendent des distances focales des lentilles. Cependant, comme nous l'avons vu aux sections 4.2.2 et 4.2.3, pour obtenir des systèmes électroniques très performants, il faut souvent plusieurs cartes comportant plusieurs puces. Un tel système nécessite alors une source d'alimentation volumineuse, de sorte que l'ensemble peut facilement devenir aussi encombrant que le système optique. Par contre, pour certaines applications nécessitant de petits réseaux pouvant être aisément contenus dans une seule puce, sans trop de circuiterie externe, et si la puissance consommée est compatible avec l'usage d'une petite pile pour l'alimentation, l'électronique est alors le meilleur choix possible.

Pour ce qui est du poids, comme les composants dans un système optique sont relativement distants les uns des autres pour un même volume occupé, le système optique sera beaucoup plus léger que le système électronique. En effet, il n'y a pas de raisons qui nous empêchent d'entasser les cartes d'un système électronique tant et aussi longtemps que la ventilation sera suffisante pour dissiper la chaleur.

P153385.PDF [Page: 142 of 191]

4.3.3 Coût économique et maturité

Il est très difficile de dissocier le coût économique d'un RNA avec la maturité de la technologie qu'il exploite. Le meilleur exemple est sans aucun doute la technologie optique pour laquelle la maturité est loin d'être atteinte. Certains composants des architectures optiques, comme les TVCL, sont déjà utilisés dans une foule d'applications commerciales comme les projecteurs vidéo, par exemple. L'utilisation de tels composants pourrait contribuer à rentabiliser le coût des architectures optiques. Malheureusement, ces composants ne rencontrent souvent pas les exigences de performance requises par les architectures neuronales. Des composants spécialisés doivent alors être créés, ce qui conduit à des coûts élevés. Il faudra probablement attendre encore quelques années avant que les grands manufacturiers de composants optiques comme NTT démontrent un intérêt plus prononcé pour la recherche en RNA.

Contrairement à la situation pour l'optique, les concepteurs de RNA électroniques ne dépendent pas des grandes compagnies pour rentabiliser leur produit. Ils doivent tout simplement (!) avoir accès à une fonderie de silicium et à de bons outils de conception assistée par ordinateur. Notons que le coût économique d'un RNA dépend en partie du coût des composants (coût de fabrication) mais aussi du coût de conception. À cet égard, les concepteurs de RNA électroniques numériques sont choyés, car ils jouissent d'une foule d'outils de conception de haut niveau tel le langage VHDL. Ces outils contribuent grandement à réduire le temps de conception et donc réduire les coûts. À l'inverse, les concepteurs de RNA analogiques n'ont pas accès à autant d'outils. De plus, il est bien connu que la validation des circuits analogiques est plus difficile que celle des circuits numériques. Cependant, les architectures analogiques exploitent davantage le parallélisme inhérent des résaux de neurones. La tolérance aux defectuosités tend donc à être meilleure. Ceci se traduit en bout de ligne par un rendement supérieur et donc un coût moindre.

P153385.PDF [Page: 143 of 191]

Finalement, le facteur le plus important pour réduire le coût global des RNA, et ce, peu importe la technologie, c'est la *vente de masse*. Pour que la vente de masse soit possible, il est impératif que les RNA soient mis en application pour résoudre des problèmes de tous les jours. Pour l'instant, la science des RNA est encore à un stade de recherche théorique et de développement de nouveaux algorithmes.

En conclusion, dans ce chapitre, nous avons confronté les deux technologies de réalisation de RNA, soit l'optique et l'électronique. Nous nous sommes basés, lors de notre analyse, sur trois grandes classes de critères: les critères de performance, les critères technologiques et les critères d'ordre pratique. Les grandes lignes de notre analyse seront reprises en détail dans la conclusion de ce rapport. Le chapitre suivant traite de la réalisation du réseau ART1 à l'aide de la technologie ITGE.

CHAPITRE V ÉTUDE DE CAS: RÉALISATION DU RÉSEAU ART1

Nous avons vu dans les chapitres précédents qu'une caractéristique intéressante d'un neuro-ordinateur ou d'une puce neuronale dédiée est sa *généralité*, c'est-à-dire sa capacité de pouvoir émuler plusieurs architectures neuronales. Bien entendu, la généralité d'une architecture entraîne toujours une complexité de design accrue et des performances souvent moindres que celles d'une architecture dédiée. Heureusement, il existe des applications qui ne nécessitent pas de généralité, mais qui, en revanche, requièrent des performances élevées. Plusieurs de ces applications sont militaires et on en retrouve un sous-ensemble en Mesures de Soutien Électronique (MSE). Au Centre de Recherche pour la Défense à Ottawa, le CRDO, une des applications en MSE est la mise en grappes de signaux radars. L'utilisation de réseaux de neurones pour accomplir cette tâche est présentement envisagée au CRDO. Le but de ce chapitre est donc d'étudier le potentiel d'un réseau en particulier, le réseau ART1 de Carpenter et Grossberg [10], dans le but ultime d'en réaliser l'algorithme en ITGE.

Le réseau ART est un excellent candidat pour ce genre d'applications, d'une part parce qu'il s'agit d'un réseau à apprentissage sans supervision et d'autre part parce que sa dynamique d'apprentissage est telle qu'elle lui permet d'apprendre de nouveaux patrons à n'importe quel moment durant son utilisation. De plus, cet apprentissage se fait sans cor-

P153385.PDF [Page: 145 of 191]

rompre la mémoire existante du réseau. Malheureusement, tel que décrit par Carpenter et Grossberg, le réseau ART1 se prête mal à la mise en œuvre ITGE, une des raisons étant qu'il nécessite une grande précision dans ses poids. La précision élevée requise est une conséquence de la loi d'apprentissage qui gouverne la modification de ses poids, la *Loi de Weber*. C'est en tentant de réduire ces besoins en précision que nous en sommes venus à proposer une nouvelle loi d'apprentissage pour le réseau ART1, la *Loi de Codage Direct*.

La première section introduit d'abord le sujet et justifie le choix de l'architecture ART1 parmi toutes les architectures ART. Le principe d'opération du réseau ART1 et de la Loi de Weber est ensuite résumé brièvement dans la section 5.2. La Loi de Codage Direct est présentée et analysée dans la section 5.3. Finalement, des résultats de simulations d'un problème de détection radar passive sont présentés à la section 5.4, permettant ainsi une comparaison entre la Loi de Codage Direct et la Loi de Weber.

5.1 Introduction

La théorie de la résonance adaptative (Adaptive Resonance Theory, ART) a été introduite vers la fin des années 80 par Carpenter et Grossberg [10] comme une solution au problème de la *stabilité et de la plasticité*. Ce problème peut s'énoncer comme suit: comment un réseau peut-il demeurer stable en présence de bruit et devenir plastique, ou adaptatif, en présence d'évènements importants. Jusqu'à date, plusieurs architectures dont le principe repose sur ART ont été proposées (ART1, ART2, ARTMAP, etc.). Le réseau qui nous intéresse ici est le ART1 [10] pour la simplicité de son algorithme faisant de lui un excellent candidat en vue d'une mise en œuvre ITGE [72][78]. Par ailleurs, le fait que le réseau ART1 ne traite que des patrons d'entrée binaires n'est pas en soi une limitation puisqu'il existe une multitude de codes pouvant convertir un vecteur réel en binaire tout en préservant les distances entre les éléments [19][65]. C'est pourquoi, de nos jours, les réseaux

P153385.PDF [Page: 146 of 191]

ART1 trouvent de plus en plus leur place dans des applications industrielles ou militaires [11][80].

Les réseaux ART1 font appel, lors de leur phase d'apprentissage, à une loi connue sous le nom de *Loi de Weber* ou *Loi de Codage Inverse* [10]. Un des problèmes de cette loi est que la précision requise dans les poids qu'elle ajuste croît comme le logarithme du nombre d'éléments N à l'entrée du réseau. Rao *et al.* ont démontré que le nombre de bits b requis pour ces poids doit satisfaire l'inégalité $b \geq 3\log_2 N$ [72]. Conséquemment, les auteurs ont conclu que «cette loi se prête mal aux réalisations ITGE» [72]. En effet, il est bien connu que le délai de propagation à travers la chaîne de retenue des modules arithmétiques d'un circuit ITGE numérique représente souvent le goulot d'étranglement de la performance.

Dans le but de réduire les besoins en précision de la loi d'apprentissage, nous proposons une version modifiée de la Loi de Codage Inverse, que nous appelons la *Loi de Codage Direct*. La principale distinction entre les deux lois est qu'une division dans la Loi de Codage Inverse est remplacée par une soustraction dans la Loi de Codage Direct. Nous allons démontrer qu'en plus de réduire les besoins en précision des poids, cette modification accélère la convergence du réseau en modifiant sa dynamique de recherche.

5.2 Principe d'opération du réseau ART1 et la Loi de Weber

Le réseau ART1 est un réseau de mise en grappes (clustering) sans supervision. Il est constitué d'une couche de neurones d'entrée et d'une couche de neurones de sortie, toutes deux identifiées respectivement par F1 et F2 (voir figure 5.1). Le terme "sans supervision" signifie que le réseau ne nécessite aucune aide du monde extérieur pour effectuer sa mise en grappes: il n'utilise que l'information contenue dans les données pour former les différentes grappes.

P153385.PDF [Page: 147 of 191]

Les patrons d'entrée sont présentés aux neurones de la couche F1. Ces derniers propagent le patron sur tous les neurones de la couche F2. Suite à la présentation d'un patron I, tous les neurones de la couche F2 calculent leur *entrant net*, défini par l'équation (5.1) suivante:

$$T_j = \sum_{i=1}^{N} I_i w_{ij}, \tag{5.1}$$

où w_{ij} dénote le poids bas-vers-haut (BH) reliant le neurone F1 i au neurone F2 j. Les neurones de la couche F2 entrent alors en compétition pour déterminer lequel d'entre eux

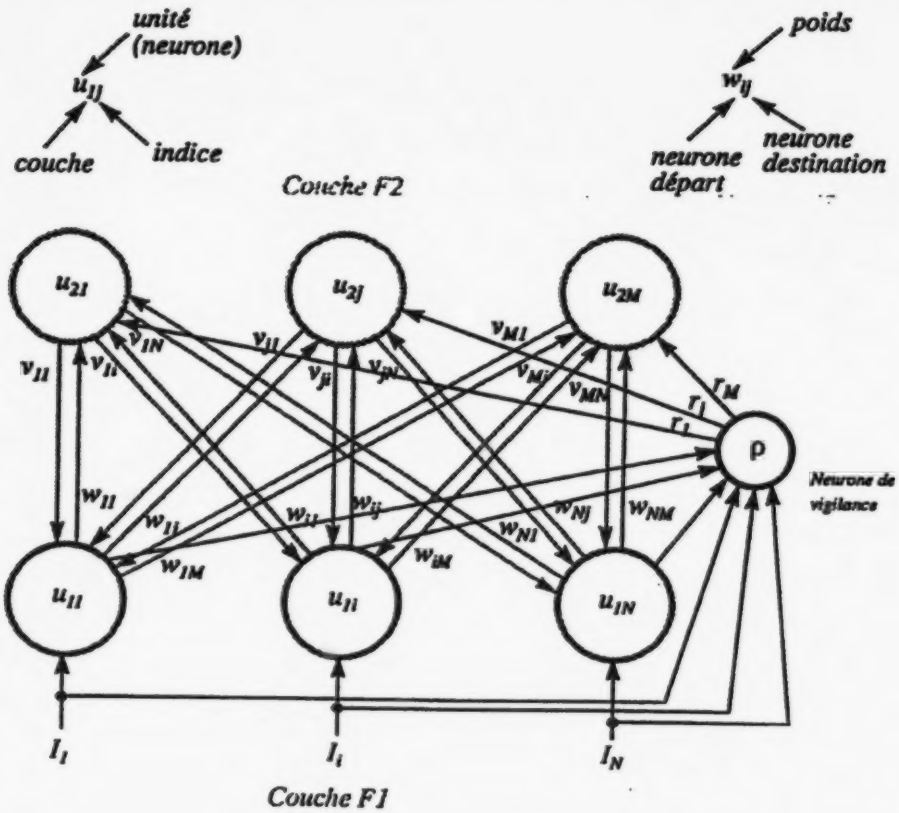


Figure 5.1: Schéma du réseau ART1.

P153385.PDF [Page: 148 of 191]

possède l'entrant net le plus élevé. Le neurone gagnant, d'indice J, envoie alors son prototype $\mathbf{V}_J$ à la couche F1 pour que le calcul du *test de vigilance* prenne place. Le prototype $\mathbf{V}_J$ correspond aux N poids binaires haut-bas (HB) v_{Ji}, $i = 1, 2, ..., N$, émanant du neurone F2 d'indice J vers les N neurones de la couche F1.

Le but du test de vigilance est de calculer la similarité entre le prototype $\mathbf{V}_J$ et le patron d'entrée $\mathbf{I}$, pour ensuite la confronter à un seuil ρ appelé *seuil de vigilance* ou *critère de vigilance* selon l'équation

$$\frac{\|\mathbf{I} \cap \mathbf{V}_J\|}{\|\mathbf{I}\|} \geq \rho, \tag{5.2}$$

où $\mathbf{I} \cap \mathbf{V}_J$ est l'intersection (ET-binaire) entre $\mathbf{I}$ et $\mathbf{V}_J$ et $\|\cdot\|$ est l'opérateur "norme", c'est-à-dire $\|\mathbf{X}\| = \sum_{i=1}^{N} x_i$.

Si le prototype réussit le test de vigilance, on dira que le neurone J entre en *résonance*, période durant laquelle il ajuste ses poids $\mathbf{V}_J$ et $\mathbf{W}^{(J)}$ ($\mathbf{W}^{(J)} = \{w_{iJ} | i = 1, 2, ...N\}$) pour *apprendre* le patron $\mathbf{I}$ selon les lois d'apprentissage définies plus loin. Sinon, le réseau désactivera temporairement le neurone J et engagera une nouvelle compétition pour déterminer un nouveau prototype candidat. Ce processus de recherche itératif se poursuivra jusqu'à ce qu'il y ait résonance ou jusqu'à ce qu'une nouvelle classe soit créée en assignant le patron $\mathbf{I}$ à un neurone F2 non-commis. Le réseau est alors prêt à classifier un nouveau patron d'entrée.

La désactivation des neurones F2 est contrôlée par le *neurone de vigilance* (voir figure 5.1). Ce dernier émet un signal r_j destiné à chacun des neurones F2. Si $r_j = 1$, le neurone j sera désactivé jusqu'à la présentation d'un nouveau patron d'entrée. À la fin du cycle de reconnaissance, tous les signaux r_j sont initialisés à zéro.

P153385.PDF [Page: 149 of 191]

La mise à jour des poids $\mathbf{V}_j$ et $\mathbf{W}^{(J)}$, suite à une résonance, est régie d'une part par la *loi d'apprentissage des prototypes* (template learning rule, c.f. [10]) définie par

$$\mathbf{V}_j' = \mathbf{I} \cap \mathbf{V}_j , \tag{5.3}$$

et d'autre part par la *Loi de Weber* (Weber Law Rule), c.f. [10], définie par

$$\mathbf{W}'^{(J)} = \frac{\alpha(\mathbf{I} \cap \mathbf{V}_j)}{\beta + \|\mathbf{I} \cap \mathbf{V}_j\|} = \frac{\alpha \mathbf{V}_j'}{\beta + \|\mathbf{V}_j'\|} , \tag{5.4}$$

où le "prime" dans cette équation dénote la version mise-à-jour du vecteur de poids et α et β sont des constantes. Pour alléger la notation et, comme nous le verrons plus tard, sans perte de généralité, nous posons $\alpha = 1$. De plus, selon Carpenter et Grossberg, et dans le but d'établir certaines propriétés énoncées plus bas, $\beta = 1/(\|\mathbf{I}\|_{max} - 1)$ [10].

Au début de chaque cycle de reconnaissance, les poids $\mathbf{W}^{(J)}$ d'un neurone F2 d'indice j sont reliés à ses poids $\mathbf{V}_j$ par

$$\mathbf{W}^{(J)} = \frac{\mathbf{V}_j}{\beta + \|\mathbf{V}_j\|} , \tag{5.5}$$

qui est essentiellement l'équation (5.4) mais sans la notation «prime» pour indiquer qu'il s'agit d'un nouveau cycle de reconnaissance. Donc, en utilisant les équations (5.1) et (5.5), l'entrant net se réduit à:

$$T_j = \sum_{i=1}^{N} I_i w_{ij} = \sum_{i=1}^{N} \frac{I_i v_{ji}}{\beta + \|\mathbf{V}_j\|} = \frac{1}{\beta + \|\mathbf{V}_j\|} \sum_{i=1}^{N} I_i v_{ji} = \frac{\|\mathbf{I} \cap \mathbf{V}_j\|}{\beta + \|\mathbf{V}_j\|} . \tag{5.6}$$

Cette fonction de I et de $\mathbf{V}_j$, qui porte le nom de *fonction d'ordonnancement*, est cruciale au fonctionnement du réseau ART1. En effet, cette fonction caractérise l'ordre dans lequel les prototypes appris seront soumis au test de vigilance. Définissons à présent les trois situations possibles qui peuvent se produire lorsqu'un patron d'entrée I est présenté au réseau: le prototype $\mathbf{V}_j$ d'un neurone F2 d'indice j donné peut être 1) un *sous-ensemble*

P153385.PDF [Page: 150 of 191]

de **I**, i.e. $\mathbf{V}_j \subseteq \mathbf{I}$, 2) un *sur-ensemble* de **I**, i.e. $\mathbf{V}_j \supset \mathbf{I}$, ou 3) un *ensemble mixte* de **I** [10]. En utilisant ces définitions, l'ordre de recherche parmi les prototypes appris peut être caractérisé par quatre propriétés:

1. Les sous-ensembles sont recherchés d'abord, en ordre décroissant de grandeur.
2. S'il existe d'autres prototypes et que ces derniers sont des sur-ensembles, alors la recherche s'arrête au plus petit sur-ensemble.
3. S'il existe d'autres prototypes et que ces derniers sont des sur-ensembles et des ensembles mixtes, alors la recherche s'effectue parmi les prototypes mixtes qui satisfont $\|\mathbf{I} \cap \mathbf{V}_j\| / \|\mathbf{V}_j\| > \|\mathbf{I}\| / \|\mathbf{V}_J\|$, où $\mathbf{V}_j$ est un prototype mixte et $\mathbf{V}_J$ est le plus petit sur-ensemble, en ordre décroissant du ratio $\|\mathbf{I} \cap \mathbf{V}_j\| / \|\mathbf{V}_j\|$.
4. S'il existe d'autres prototypes et que ces derniers sont uniquement des ensembles mixtes, alors la recherche s'effectue parmi ceux qui satisfont la relation $\|\mathbf{V}_j \cap \mathbf{I}\| / (\beta + \|\mathbf{V}_j\|) > w_0 \|\mathbf{I}\|$, où w_0 est la valeur initiale des poids bas-haut des neurones non-commis. Si tous ces prototypes échouent le test de vigilance, alors un nouveau prototype identique au patron d'entrée **I** est assigné à un neurone non-commis.

La preuve de ces quatre propriétés est donnée dans la référence [10].

5.3 Loi de Codage Direct

Nous pouvons attribuer les besoins de précision élevés de la Loi de Weber à la division de l'équation (5.4). En effet, afin de respecter la relation d'ordre impliquée par les opérations mathématiques, la division de deux mots, en arithmétique à virgule fixe, nécessite une précision à la sortie plus élevée que la précision des mots à l'entrée. La quantification du quotient peut, si elle n'est pas suffisamment élevée, résulter en une erreur d'arrondi. Dé plus, le cumul de ces erreurs d'arrondi dans la sommation (5.1) contribue à perturber l'ordre de recherche des prototypes.

P153385.PDF [Page: 151 of 191]

Une analyse de premier ordre illustre bien la conséquence de la division dans (5.4) sur la précision des poids. Pour être en mesure de distinguer tous les poids, la précision doit être suffisamment élevée pour distinguer n'importe quelle paire de valeurs d'entrée (i.e. $\|\mathbf{V}_j\|$). Si N est le nombre de bits des prototypes, et donc de l'entrée, alors $\|\mathbf{V}_j\|$ peut assumer $N+1$ valeurs distinctes variant de 0 à N. Soit $N = 2^n - 1$, où n est le nombre de bits requis pour représenter la norme. Ainsi, $n = \lceil \log_2(N+1) \rceil$ bits seront nécessaires pour représenter la norme. Le pire cas, quand il s'agit de préserver l'ordre suite à une division, implique la distinction des poids correspondants à la plus grande et la deuxième plus grande valeur de $\|\mathbf{V}_j\|$, soit $2^n - 1$ et $2^n - 2$. Le nombre de bits de précision b requis pour les poids, en supposant que $\alpha = 1$ et $\beta \ll N$ (voir équation (5.4)), est donné par:

$$\frac{1}{2^b} < \frac{1}{2^n - 2} - \frac{1}{2^n - 1} \cong \frac{1}{2^{2n}} \Rightarrow b > 2n. \tag{5.7}$$

Ainsi, la division dans la Loi de Weber double le nombre de bits requis pour distinguer toutes les paires de poids. Notez qu'un autre inconvénient de la division est son coût de mise en œuvre élevé, lorsque comparé à celui d'une addition ou d'une soustraction.

En réponse aux problèmes de coût élevés de la division et à l'augmentation systématique de la précision requise dans les calculs, nous introduisons à présent une nouvelle loi d'apprentissage qui, contrairement à la loi de Weber, ne nécessite pas de division. D'après (5.4), lorsque $\|\mathbf{V}_j\|$ est élevé, w_{ij} est faible et vice-versa. Un comportement similaire peut être obtenu par une soustraction, et la règle d'apprentissage s'énonce alors comme suit:

$$w_{ij}' = \begin{cases} \Psi - \|\mathbf{V}_j'\| & \text{si } v_{ji}' = 1 \\ 0 & \text{autrement,} \end{cases} \quad \text{ou encore } \mathbf{W}'^{(J)} = \mathbf{V}_j' (\Psi - \|\mathbf{V}_j'\|) \tag{5.8}$$

où Ψ est un constante positive dont la valeur sera déterminée ultérieurement. L'équation (5.8) énonce la *Loi de Codage Direct*. Notez que pour une soustraction, le nombre de bits

P153385.PDF [Page: 152 of 191]

requis pour le résultat augmente d'un bit seulement comparativement au nombre de bits des termes soustraits. La Loi de Codage Direct permet par conséquent de distinguer les poids avec beaucoup moins de bits que la Loi de Weber traditionnelle.

L'ordre de recherche avec lequel les prototypes sont balayés selon la Loi de Codage Direct est quelque peu différent de celui selon la Loi de Weber. Dans ce qui suit, nous tenterons de caractériser cet ordre de recherche et de déterminer une valeur adéquate pour la constante Ψ. En particulier, il sera démontré que les propriétés 1 et 2 décrites dans la section précédente peuvent être reproduites. Pour ce qui est des propriétés 3 et 4, l'ordre de recherche des prototypes mixtes est complexe et il ne sera pas analysé dans le cadre de ce mémoire. Mentionnons cependant que les résultats de simulation présentés plus loin indiquent que cet ordre est différent et qu'il conduit à des vitesses d'apprentissage supérieures avec la Loi de Codage Direct.

Considérons tout d'abord l'ordre de recherche des prototypes sous-ensembles. Pour ces derniers, $\|\mathbf{V}_j \cap \mathbf{I}\| = \|\mathbf{V}_j\|$. Par conséquent, l'entrant net T_j devient:

$$T_j = \|\mathbf{V}_j\| (\Psi - \|\mathbf{V}_j\|) \text{ pour les prototypes sous-ensembles.} \tag{5.9}$$

La propriété 1 nous dit que tous les prototypes sous-ensembles doivent être balayés en ordre décroissant de grandeur. Autrement dit, nous devons nous assurer que $\forall i, j$, si $\|\mathbf{V}_i\| > \|\mathbf{V}_j\|$, alors $T_i > T_j$. En utilisant (5.9) pour développer l'inégalité précédente, nous obtenons:

$$\|\mathbf{V}_i\| (\Psi - \|\mathbf{V}_i\|) > \|\mathbf{V}_j\| (\Psi - \|\mathbf{V}_j\|). \tag{5.10}$$

En isolant Ψ, on trouve $\Psi > \|\mathbf{V}_i\| + \|\mathbf{V}_j\|$. Dans le cas limite, $\|\mathbf{V}_i\| = N$ et $\|\mathbf{V}_j\| = N - 1$, et l'inégalité devient

$$\Psi > 2N - 1. \tag{5.11}$$

P153385.PDF [Page: 153 of 191]

Cette inégalité nous assure donc que tous les prototypes sous-ensembles seront balayés en ordre décroissant de grandeur.

La propriété 1 nous dit également que tous les prototypes sous-ensembles doivent être balayés avant les prototypes sur-ensembles ou les mixtes. Pour les prototypes sur-ensembles, $\|V_j \cap I\| = \|I\|$ et l'entrant net se réduit à

$$T_j = \|I\| (\Psi - \|V_j\|) \text{ pour les prototypes sur-ensembles,} \qquad (5.12)$$

qui, en fonction de $\|V_j\|$, décrit l'équation d'une droite de pente $-\|I\|$. La figure 5.2 illustre la famille de courbes d'ordonnancement pour différentes valeurs de $\|I\|$, $N = 16$ et $\Psi = 32$. La fonction d'ordonnancement T est tracée en fonction de $\|V\|$ et $\|I\|$. L'axe horizontal est la valeur de $\|V\|$. Remarquez que, bien que les courbes soient tracées comme si elles étaient continues, elles ne sont en fait définies qu'aux valeurs entières de $\|V\|$. Pour une grandeur d'entrant $\|I\|$ donnée, si nous écartons temporairement de la discussion le cas des ensembles mixtes, la fonction d'ordonnancement assume d'abord une

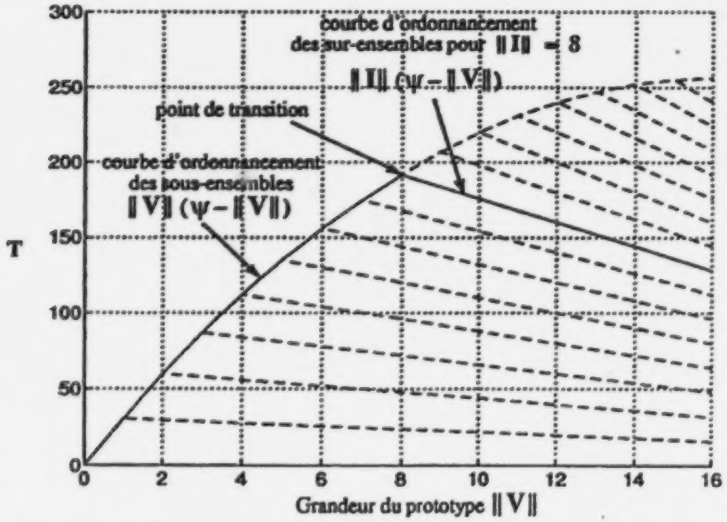


Figure 5.2: Famille de courbes d'ordonnancement pour $N = 16$ et $\Psi = 32$.

P153385.PDF [Page: 154 of 191]

progression de sous-ensemble (équation (5.9)) pour des valeurs de $\|V\|$ inférieures à $\|I\|$, puis une progression de sur-ensemble (équation (5.12)) pour des valeurs de $\|V\|$ supérieures à $\|I\|$.

Pour s'assurer que les sous-ensembles seront balayés avant les sur-ensembles et les ensembles mixtes, il suffit d'ajouter un biais positif Θ_j à l'entrant net des neurones dont le prototype est un sous-ensemble du patron d'entrée. Ce biais doit être supérieur à la valeur crête de la fonction d'ordonnancement des sur-ensembles et des ensembles mixtes. En supposant que $\Psi > 2N - 1$, après quelques manipulations, nous obtenons

$$\max(T_j) = \begin{cases} N^2 - 2N + 1 & \text{pour les sur-ensembles} \\ N^2 - 2N & \text{pour les ensembles mixtes.} \end{cases} \tag{5.13}$$

Le biais sera donc supérieur au maximum de ces deux valeurs, soit $N^2 - 2N + 1$. L'entrant net de l'équation (5.1) devient alors:

$$T_j = \sum_{i=1}^{N} I_i w_{ij} + \Theta_j, \tag{5.14}$$

où $\Theta_j > N^2 - 2N + 1$ pour les prototypes sous-ensembles et $\Theta_j = 0$ autrement. Cette relation, combinée avec l'équation (5.11) forment les conditions nécessaires et suffisantes pour que la propriété 1 soit préservée.

La propriété 2 spécifie que, si après avoir balayé tous les prototypes sous-ensembles, il existe d'autres prototypes et que ces derniers sont des sur-ensembles, alors la recherche s'arrête au plus petit sur-ensemble. Cette propriété est assurée par la Loi de Codage Direct par le simple fait que les courbes d'ordonnancement des sur-ensembles sont des droites de pentes négatives.

P153385.PDF [Page: 155 of 191]

5.4 Résultats de simulations

Dans cette section, nous étudions le potentiel des réseaux ART pour une application de compression de données en mesures de soutien électronique (MSE). Le rôle des MSE est d'intercepter, de localiser, d'enregistrer et d'analyser des signaux radars pour la surveillance militaire et, si nécessaire, d'entreprendre des contre-mesures [74][35]. La nécessité de la compression de données est en partie due aux grandes densités de signaux qu'on retrouve dans certaines bandes qui, dans plusieurs cas, peuvent atteindre des débits d'un million d'impulsions radar à la seconde. Par conséquent, la mise en grappes (*clustering*) des impulsions en fonction de l'émetteur duquel ils sont issus au tout début du traitement peut réduire considérablement le temps de traitement global du système ESM. Pour nous convaincre de cette affirmation, un ensemble de 800 vecteurs représentant des impulsions a été obtenu du Centre de Recherche pour la Défense Ottawa (CRDO). Cet ensemble de vecteurs a été bâti suite à une collecte de données entreprise par le CRDO et renferme des impulsions radar en provenance de 12 émetteurs différents.

Chaque vecteur représente 8 échantillons complexes extraits du spectre d'une impulsion radar. Ces 8 échantillons sont alors convertis en patrons binaires, compatibles avec les réseaux ART1, en encodant l'amplitude et la phase à l'aide d'un code thermomètre [65] et d'un code rampant (*creeping code*) [19], respectivement. Ces deux codes sont illustrés à la figure 5.3a et 5.3b respectivement. Cette technique d'encodage produit des

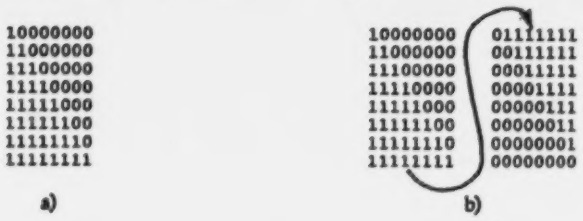


Figure 5.3: Code a) thermomètre et b) rampant.

P153385.PDF [Page: 156 of 191]

patrons de $N = 8 \times 10 + 8 \times 5 = 120$ bits. Les deux codes utilisés ont l'avantage de préserver la distance après l'encodage. Autrement dit, si la distance entre deux valeurs est faible, alors la distance entre les deux mots codes (distance de Hamming) sera également faible, et vice-versa. De plus, le code rampant possède l'avantage de préserver la propriété cyclique inhérente à la phase (i.e. la fait que $2\pi - \varepsilon$ soit près de 0).

Les patrons ont été présentés à cinq réseaux ART1 dans le but d'effectuer leur mise en grappes. Pour la loi d'apprentissage des poids bas-vers-haut, quatre des cinq réseaux font appel à la loi de Weber avec des niveaux de quantification dans les poids de 28, 14, 10 et 7 bits. Le cinquième réseau, quant à lui, utilise la Loi de Codage Direct avec un niveau de quantification de 7 bits.

Pour chaque simulation, les 800 patrons d'entrée sont présentés à un des réseaux et la mise en grappes est caractérisée. Suite à la présentation de chaque patron, le réseau ART1 entreprend une recherche itérative selon laquelle chaque prototype en mémoire est comparé avec le patron d'entrée, dans un ordre spécifique, jusqu'à ce que le test de vigilance soit positif. Chaque itération de ce processus porte le nom d'*itération de recherche*. La *convergence* est l'atteinte de la stabilité, définie comme l'état du réseau pour lequel tous les patrons d'entrée ont été mémorisés et les poids demeurent stables. Nous définissons alors le *temps de convergence* comme le temps (mesuré en itérations de recherche) requis pour atteindre la convergence. Cette dernière métrique mesure l'effort de calcul d'une mise en œuvre séquentielle du réseau ART1.

Le temps de convergence d'un réseau pour un ensemble donné de patrons dépend évidemment de l'ordre avec lequel ces patrons sont présentés. Afin d'obtenir la tendance moyenne du temps de convergence, les patrons sont présentés dans un ordre aléatoire, et la

moyenne du temps de convergence est calculée à partir des résultats de 10 simulations, soit 10 ordres aléatoires de présentation différents.

La figure 5.4 illustre le temps de convergence de chacun des cinq réseaux ART1 en fonction des valeurs du paramètre de vigilance variant de 0.1 à 1.0. Si nous examinons les réseaux qui utilisent la loi de Weber, il semble que le temps de convergence croît graduellement au fur et à mesure que le niveau de quantification des poids est diminué. Par exemple, le réseau avec des poids de 7 bits prend de 4 à 14 fois plus de temps à atteindre la convergence, comparativement au réseau dont les poids sont quantifiés à 28 bits, pour des valeurs de ρ comprises entre 0.4 et 0.9. Nous avions anticipé un tel comportement, étant donné que la quantification des poids a pour effet de perturber l'ordre de recherche, augmentant ainsi le temps de convergence.

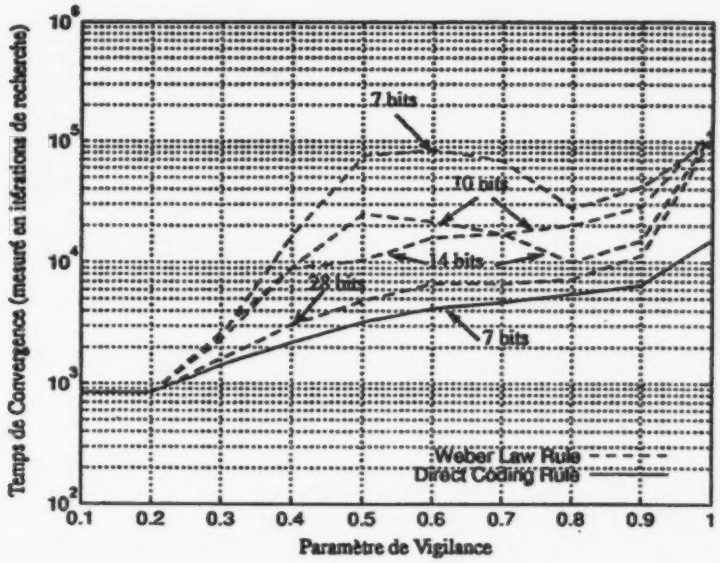


Figure 5.4: Temps de convergence, mesurés en itérations de recherche, pour les cinq réseaux ART1.

P153385.PDF [Page: 158 of 191]

La figure 5.4 illustre aussi le fait que le réseau qui fait appel à la Loi de Codage Direct converge plus vite que tous les autres réseaux qui font appel à la loi de Weber. Définissons le *gain de vitesse* que procure la loi de Codage Direct sur la Loi de Weber comme étant $100 \times (t_{LW} - t_{LCD}) / t_{LCD}$, où t_{LCD} et t_{LW} sont les temps de convergence pour les réseaux utilisant les deux lois respectivement. Tous nos résultats de gain de vitesse seraient donc rapportés à la performance de la Loi de Codage Direct, qui s'est avérée la plus rapide de toutes. Si nous calculons un gain de vitesse moyen entre $\rho = 0.3$ et 1.0, la Loi de Codage Direct avec des poids de 7 bits est 45% plus rapide que la Loi de Weber avec des poids de 24 bits. Lorsque la précision des poids pour la loi de Weber est graduellement réduite à 14, 10 et 7 bits, le gain de vitesse de la Loi de Codage Direct augmente à 253%, 390% et 980%, respectivement. Il est à noter que ces valeurs de gain nous permettent de tirer des conclusions uniquement sur la vitesse du processus de recherche itérative, et non sur la vitesse globale du réseau car, selon la loi d'Amdahl, le gain de vitesse doit être pondéré par la proportion du temps d'exécution durant lequel il est en vigeur. Un exemple d'une autre tâche dont le temps d'exécution est grand est le calcul du produit scalaire de l'équation (5.1) pour tous les neurones F2.

La convergence rapide qu'entraîne la Loi de Codage Direct s'explique en partie par sa grande tolérance à la quantification des poids. Cependant, le fait que cette dernière permette une convergence plus rapide que la Loi de Weber est surprenant. Ce phénomène s'explique probablement par l'incidence de l'ordre de recherche parmi les prototypes de type mixte. En effet, il est possible que la Loi de Codage Direct effectue sa recherche dans un plus petit ensemble des prototypes mémorisés, et que ces derniers sont de meilleurs candidats à réussir le test de vigilance.

Une comparaison des cinq réseaux ne saurait être complète sans la vérification de la qualité des mises en grappes produites. La technique à laquelle nous faisons appel est

P153385.PDF [Page: 159 of 191]

basée sur la mesure de similarité entre les mises en grappes, telle que proposée par Rand [71] et décrite par Anderberg [1].

Une *mise en grappes* est une partition de p patrons d'entrée en k grappes. Pour une mise en grappes donnée, définissons x comme étant le nombre total de paires de patrons d'entrée appartenant à la *même* grappe et y, le nombre total de paires de patrons appartenant à des grappes *différentes*. Alors $x+y$ représente le nombre total de combinaisons de 2 éléments parmi p, soit $p(p-1)/2$. Deux mises en grappes A et B peuvent être comparées à l'aide d'un tableau de contingence comme celui du tableau 5.1:

Tableau 5.1: Tableau de contingence.

		Mise en grappes B	
		même	différentes
Mise en grappes A	même	a	c
	différentes	d	b

Par exemple, d représente le nombre de paires de patrons qui ont été regroupés dans la même grappe par la partition B, mais placés dans des grappes différentes par la partition A. La mesure de similarité entre deux mises en grappes est:

$$S(\mathrm{A},\mathrm{B}) = \frac{a+b}{a+b+c+d}. \tag{5.15}$$

Cette équation donne un nombre entre 0 et 1: plus $S(\mathrm{A},\mathrm{B})$ tend vers 1, plus les deux mises en grappes se ressemblent et plus $S(\mathrm{A},\mathrm{B})$ tend vers 0, moins elles se ressemblent.

Étant donné que nous connaissons *a priori* les émetteurs de chacune des impulsions, nous pouvons utiliser cette information pour quantifier la qualité des mises en grappes. Il est à noter qu'un système MSE ne dispose normalement pas de cette information extrinsèque. En utilisant l'équation (5.15) pour comparer une partition A avec une partition de référence R, $S(\mathrm{A},\mathrm{R})$ peut alors être interprété comme un *score*, plutôt qu'une similarité.

P153385.PDF [Page: 160 of 191]

La partition de référence est simplement celle produite en utilisant l'information *a priori* sur les émetteurs.

En considérant que, en MSE radar, $k \ll p$ et donc $a+d \ll b+c$, les paramètres du tableau 5.1 peuvent être normalisés par rapport aux dimensions des grappes de la partition R. Ainsi, après avoir divisé a et d par $(a+d)$, et b et c par $(b+c)$ dans (5.15), le score devient alors

$$S'(\mathrm{A},\mathrm{R}) = \frac{1}{2}\left(\frac{a}{a+d}+\frac{b}{b+c}\right) . \tag{5.16}$$

La valeur du score se situe toujours entre 0 et 1, sauf que les paires appartenant à la même grappe sont maintenant traitées séparément des paires appartenant à des grappes différentes. En particulier, si la partition A correspond à une seule grappe de dimension p ou p grappes de dimension 1, alors $S'(\mathrm{A},\mathrm{R}) = 0.5$, peu importe la partition de référence R. Il est à noter que $S'(\mathrm{A},\mathrm{R})$ est défini uniquement si la partition de référence R contient au moins deux grappes ($(b+c)>0$) et deux patrons dans une de ces grappes ($(a+d)>0$).

Pour quantifier la qualité des mises en grappes de chaque réseau, nous calculons la moyenne des scores des 10 partitions produites par le réseau. La figure 5.5 illustre les scores moyens en fonction du paramètre de vigilance ρ. À première vue, les scores des cinq réseaux semblent similaires. Une observation plus poussée révèle que les réseaux faisant appel à la Loi de Weber, dont les poids sont quantifiés à 24-bits, et à la Loi de Codage Direct, dont les poids sont quantifiés à 7 bits, permettent d'obtenir des scores légèrement plus élevés que les autres réseaux, pour des valeurs de ρ entre 0.4 et 1.0.

En conclusion, une nouvelle loi d'apprentissage, la Loi de Codage Direct a été proposée en remplacant la division dans l'équation de mise à jour des poids bas-haut des

P153385.PDF [Page: 161 of 191]

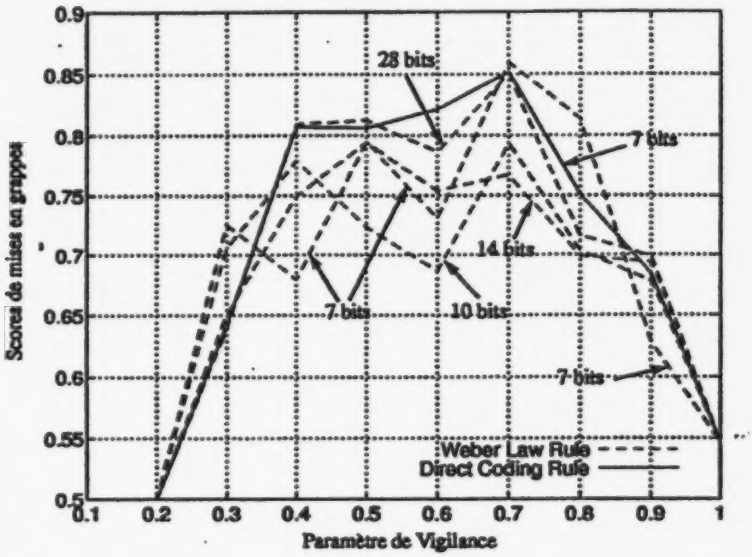


Figure 5.5: Scores obtenus pour les mises en grappes des cinq réseaux ART1.

réseaux ART1 par une soustraction. Cette nouvelle loi a été comparée à la Loi de Weber pour une application de mise en grappes de signaux radar, un important prétraitement en mesures de soutien électronique. Des résultats de simulation par ordinateur démontrent que la nouvelle loi nécessite considérablement moins de précision dans les poids, requiert moins d'itérations de recherche pour atteindre la convergence et est moins dispendieuse à mettre en œuvre en ITGE que la loi originale. Ces résultats nous laissent anticiper que la Loi de Codage Direct pourrait être utile dans un certain nombre d'applications industrielles ou militaires.

CONCLUSION

Ce projet de maîtrise comportait deux volets précis. Dans le cadre du premier volet, nous avons réalisé une étude comparative des principales technologies de réalisation des réseaux de neurones artificiels. Ces technologies sont l'optique et l'électronique. Dans le second volet, nous avons étudié le cas de la réalisation du réseau *Adaptive Resonance Theory 1* (ART1) en technologie électronique. Cette étude nous a conduit à apporter quelques modifications à l'algorithme ART1 dans le but ultime de faciliter sa mise en œuvre. Voici donc les principales conclusions de notre ouvrage, en commençant par l'étude comparative.

Premier volet: Étude comparative

Sans doute la conclusion la plus importante que l'on puisse tirer de cette étude est qu'aucune des deux technologies ne surclasse l'autre complètement. L'objectif principal de ce projet était justement de bien clarifier selon quels critères la supériorité d'une technologie pouvait être prononcée.

En matière de performance, la technologie optique possède une longueur d'avance sur l'électronique, mais seulement pour une classe restreinte d'architectures: les architectures à corrélateurs optiques. Ces architectures tirent leur supériorité de l'utilisation optimale du principal bienfait de la technologie optique, notamment son pouvoir d'interconnexion.

Notre conclusion est que l'optique est le choix intelligent pour des applications où l'électronique n'a aucune chance de rivaliser. Il n'est d'ailleurs pas surprenant que pour des applications où l'optique tente d'imiter l'électronique (dans les architectures à processeurs algébriques optiques, par exemple) le gain de performance de l'optique versus l'électronique est souvent minime ou même parfois négatif.

Une vérité partagée par l'ensemble de la communauté scientifique est que, à ce jour, nous ne savons toujours pas ce qu'est *réellement* un réseau de neurones artificiels. Donc, comment pouvons-nous choisir objectivement une technologie de réalisation si nous ne savons même pas *a priori* ce que nous voulons réaliser? C'est, à notre avis, sur cette base que l'électronique numérique a bâti sa popularité. En effet, la grande flexibilité de l'électronique numérique facilite considérablement la conception d'architectures génériques pouvant émuler une multitude de paradigmes neuronaux. La flexibilité permet également d'atteindre une plus grande clientèle cible lors de la commercialisation, ce qui explique pourquoi la majeure partie des réalisations commercialement disponibles sont numériques plutôt qu'analogiques ou optiques.

Notons cependant que la supériorité de l'électronique numérique sur le plan commercial pourrait bientôt être atténuée. La tendance actuelle chez les concepteurs de machines neuronales est de concevoir de plus en plus leurs architectures avec une préoccupation accrue pour l'application. Cette méthodologie est bénéfique pour plusieurs raisons. D'abord, les chances de commercialisation sont beaucoup plus grandes si, bien entendu, l'application est utile en soi. De plus les besoins en exactitude dans les calculs peuvent être mieux évalués si l'application est connue.

Malgré cette tendance, l'apparition de RNA optiques sur le marché n'est pas pour demain, et ce, en raison de la faible maturité de cette technologie. En effet, n'oublions pas

P153385.PDF [Page: 164 of 191]

que l'optique doit tenir tête à l'électronique, dont la maturité dépasse les 50 ans. Bon nombre des coûts élevés en optique sont la conséquence de la faible maturité de cette technologie. De plus, les sommes injectées dans la recherche en électronique sont considérablement plus importantes que celles injectées dans la recherche en optique, ce qui contribue au ralentissement de l'innovation en optique.

De paire avec le facteur maturité, on ne doit pas non plus sous-estimer le facteur *confiance*. Ainsi, il est fort probable qu'à performances égales et à coûts comparables, une architecture électronique sera choisie en faveur d'une architecture optique, strictement sur la base de la confiance. De plus, plusieurs problèmes de fragilité et de sensibilité aux vibrations mécaniques devront être résolus en optique avant d'entreprendre la commercialisation.

L'électronique a également plusieurs longueurs d'avance sur l'optique en matière d'encapsulation. De nos jours, la fabrication de puces ITGE dédiées est devenue pratique courante. L'optique intégrée, par ailleurs, en est encore à ses premiers balbutiements. Les dimensions encombrantes des RNA optiques restreignent donc l'applicabilité de l'optique lorsque la miniaturisation est un critère déterminant. Cependant, les performances d'architectures électroniques miniaturisées seront modérées. Pour des systèmes à hautes performances, la mise à l'échelle est de rigueur. Les dimensions des architectures électroniques et optiques deviennent alors comparables.

En ce qui a trait à la puissance, l'électronique ne présente pas d'avantages marqués sur l'optique. Cependant, l'électronique analogique en mode infraseuil est de loin la solution la plus économique en puissance. Ces architectures ont donc de fortes chances de se retrouver dans les satellites et les téléphones cellulaires portatifs de demain.

Deuxième volet: Étude de cas

Dans le deuxième volet, nous nous sommes intéressés à la réalisation du réseau *Adaptive Resonance Theory 1* (ART1) en électronique. Ce réseau, tel que proposé par ses auteurs, Carpenter et Grossberg, est difficilement réalisable en raison de ses besoins de précision dans les poids élevés. La source de ces problèmes provient de la loi d'apprentissage des poids bas-vers-haut du réseau, la Loi de Weber, qui nécessite le calcul d'une division. Afin de réduire les besoins en précision, nous avons proposé une nouvelle loi d'apprentissage, la Loi de Codage Direct, qui remplace la division de la Loi de Weber par une soustraction.

Outre la réduction des besoins en précision, la Loi de Codage Direct permet au réseau d'atteindre la convergence plus rapidement en modifiant le processus de recherche des prototypes appris. Nous avons appliqué cette nouvelle loi à un problème de mise en grappes de signaux radar. Le temps de convergence de cinq réseaux à été mesuré par simulation. Quatre de ces réseaux faisaient appel à la loi de Weber pour laquelle les poids étaient quantifiés à 28, 14, 10 et 7 bits, et un réseau utilisait la Loi de Codage Direct avec ses poids quantifiés à 7 bits. Les résultats de simulation ont démontré qu'à un niveau de quantification égal de 7 bits, la Loi de Codage Direct procure un gain de vitesse de convergence d'un facteur de plus de 10 par rapport à la Loi de Weber. Ces résultats nous laissent anticiper que la Loi de Codage Direct pourrait être utile dans un certain nombre d'applications industrielles ou militaires.

Recommandations

Un travail de synthèse doit, par définition, restreindre son champ d'étude. Ce mémoire ne fait pas exception à cette règle. Ainsi, quelques aspects ont dus, lors de l'effort de syn-

P153385.PDF [Page: 166 of 191]

thèse, être mis de côté. Nous résumons donc ces aspects qui, à notre avis, pourraient être considérés dans le cadre d'une étude plus poussée.

Un nombre croissant de chercheurs combattent le problème de l'interconnexion en électronique en faisant appel à de nouvelles techniques de signalisation. Ces techniques ont pour effet de simplifier la circuiterie des synapses et des interconnexions. Plutôt que de représenter un nombre par une quantité physique (en analogique) ou un code binaire (en numérique), ce nombre est représenté par un certain nombre d'impulsions. Deux approches mériteraient une attention particulière: l'approche des *impulsions codées* de Murray [61] où le nombre est représenté par une fréquence d'impulsions, comme dans notre cerveau, et l'approche de la *densité d'impulsions* de Tomberg et Kaski [77], où le nombre est représenté par une densité d'impulsions à l'intérieur d'un laps de temps prédéterminé.

Une étude plus poussée devrait également examiner davantage l'émulation d'architectures neuronales sur ordinateurs parallèles à usages généraux. De nos jours, de plus en plus de stations de travail sont parallèles. Dans l'éventualité où l'achat de matériel dédié n'est pas justifiable par l'application, l'utilisation de telles machines pour l'émulation de RNA est donc un choix économique intéressant.

En ce qui a trait à la Loi de Codage Direct, un certain nombre de notions devraient, à notre avis, être approfondies. Tout d'abord, l'ordre de recherche parmi les prototypes mixtes devrait être étudié dans le but de trouver une explication au gain de vitesse de convergence. De plus, le choix et l'impact d'un code de binarisation des données sur la performance globale du réseau devrait être précisé davantage.

Ce projet a démontré à quel point le domaine de recherche en réseaux de neurones artificiels a explosé depuis ces dernières années, principalement en raison des avancements technologiques en optique ainsi qu'en électronique. Mais il ne faut pas perdre de vue que nous n'avons atteint que la fin du début de cette nouvelle branche de recherche qui, à mon avis, est extrèmement prometteuse.

BIBLIOGRAPHIE

[1] ANDERBERG, M.R., *Cluster Analysis for Applications*, Acad. Press, (1973).

[2] ATLAS, L.E., SUZUKI, Y., «Digital Systems for Artificial Neural Networks», dans *IEEE Circuits and Devices Magazine*, pp. 20-24 (novembre 1989).

[3] BAILEY, J., HAMMERSTROM, «Why VLSI Implementations of Associative VLCNs Require Connection Multiplexing», dans IEEE 2nd Annual Conference on Neural Networks, pp. II-173--II-180, (1988).

[4] BATSELL, S. G., JONG, T.-L., WALKUP, J. F., KRILE, T., «Noise limitations in optical linear algebra processors», *Applied Optics*, vol. 29(14). 2084-2090 (1990).

[5] BLAIS, S., CRÉPEAU, J., CRESPO, J.F. et KASSEM, A., «Réseau de neurones ART intégré», Rapport final du cours ELE6305, Hiver 1993.

[6] BERGERON, A., ARSENAULT, H. H., EUSTACHE, E., GINGRAS, D., «Optoelectronic Module for Winner-Take-All Operations in Optical Neural Networks», à paraître dans *Applied Optics*.

[7] BERGERON, A., ARSENAULT, H. H., GINGRAS, D., «A Single-Rail Invariant Optical Associative Memory», à paraître dans *Applied Optics*.

[8] BOAHEN, K.A., POULIQUEN, P.O., ANDREOU, A. et JENKINS, R., «A Heteroassociative Memory Using Current-Mode MOS Analog VLSI Circuits», dans *IEEE Transactions on Circuits and Systems*, vol. 36(5) pp. 747-755 (1989).

P153385.PDF [Page: 168 of 191]

[9] BOCKER, R. P., CLAYTON, S. R., BROMLEY, K., «Electrooptical matrix multiplication using the twos complement arithmetic for improved accuracy», *Applied Optics*, vol. 22(13), pp. 2019-2021 (1983).

[10] CARPENTER, G.A., et GROSSBERG, S., «A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine», dans *Computer Vision, Graphics, and Image Processing*, vol. 37, pp. 54-115 (1987).

[11] CAUDELL, T.P., «Hybrid Optoelectronic Adaptive Resonance Theory neural processor, ART1», dans *Applied Optics*, vol. 31, no. 29, pp. 6220-6229 (1992).

[12] CAULFIELD, H. G., KINSER, J., ROGERS, S. K., «Optical Neural Networks», dans *Proceedings of the IEEE*, vol. 77(10), (1989).

[13] CAULFIELD, H. J., «Parallel N^4 weighted optical interconnections», dans *Proc. of the First IEEE Conf. on Neural Networks*, pp. III-595-III-597, (1987).

[14] CAVANAGH, J.F., *Digital Computer Arithmetic*, McGraw-Hill, 468 pp. (1984).

[15] CONTROL OPTICS, fiche technique, 13111 Brooks Dr., Suite J, Baldwin Park, CA 91706.

[16] CRESPO, J.F., LAVOIE, P. et SAVARIA, Y., «Fast Convergence with Low Precision Weights in ART1 Networks», dans *Proceedings of the International Symposium on Circuits and Systems*, vol. 6 pp. 237-240 (1994).

[17] CSPI, fiche technique, 40 Linnell Circle, Billerica, MA 01821.

[18] DAS, P. K., DeCUSATIS, C. M. «Acousto-Optic Signal Processing: Fundamentals & Applications», Artech House, 473 pp. (1991).

P153385.PDF [Page: 169 of 191]

[19] DIETMEYER, D.L., *Logic Design of Digital Systems*, 2nd edition, Allyn and Bacon, Inc., pp. 227-228.

[20] DISPLAYTECH, fiche technique, 2200 Central Avenue, Boulder, Colorado, 80301, (Avril 1993).

[21] DOUCET, M., Institut National d'Optique (Communication personnelle).

[22] ELLETT, S. A., WALKUP, J. F., KRILE, T. F., «Error-correction coding for accuracy enhancement in optical matrix-vector multipliers», dans *Applied Optics*, vol. 31(26), pp. 5642-5653 (1992).

[23] EPSON, fiche technique.

[24] FEITELSON, D. G., «Optical Computing», MIT Press, 393 pp. (1992).

[25] FELDMAN, M. R., ESENER, S., C., GUEST, C., LEE, S. H., «Comparison between optical and electrical interconnects based on power and speed considerations», dans *Applied Optics*, vol. 27(9), (1988).

[26] FUKUSHIMA, S., KUROKAWA, T. et OHNO, M., «Ferroelectric liquid-crystal spatial light modulator achieving bipolar image operation and cascadability», dans *Applied Optics*, vol. 31(32), pp. 6859-6868 (1992).

[27] FUKUSHIMA, S., KUROKAWA, T. et SAKAI, Y., «Image Encipherment Based on Optical Parallel Processing Using Spatial Light Modulators», dans *IEEE Transactions Photonics Technology Letters*, vol. 3(12), pp. 1133-1135 (1991).

P153385.PDF [Page: 170 of 191]

[28] GARY, C.K., «Matrix-vector multiplication using digital partitionning for more accurate optical computing», dans *Applied Optics*, vol. 31(29), pp. 6205-6211 (1992).

[29] GARY, C. K., «Comparison of optics and electronics for the calculation of matrix-vector products», dans *SPIE Advances in Optical Information Processing V*, vol. 1704, pp. 544-555 (1992).

[30] GEC-Marconi, fiche technique, (Décembre 1993).

[31] GOODMAN, J. W., «Introduction to Fourier Optics», McGraw-Hill, 287 pp. (1968).

[32] GOODMAN, J. W., «Optics as an Interconnect Technology», dans Optical Processing and Computing, éd. Arsenault, H. H., *et al.* , pp. 1-32.

[33] GRAF, H.P., JACKEL, L.P., HUBBARD, W.E., «VLSI Implementation of a Neural Network Model», dans *Computer*, pp. 41-49, (Mars 1988).

[34] GRAF, H.P., JACKEL, L.D., «Analog Electronic Neural Network Circuits», dans *IEEE Circuits and Devices Magazine*, pp. 44-55, (juillet 1989).

[35] GRANT, P.M. et COLLINS, J.H., «Introduction to Electronic Warfare», dans *Proc. IEE (UK)*, Part F, Vol. 129, No. 3, pp. 113-129 (1982).

[36] HAMMERSTROM, D., «A VLSI Architecture for High Performance, Low-Cost, On-Chip Learning», dans *Proc. Int. Joint Conf. on Neural Networks*, vol. II pp. 537-544, (1990).

[37] HECHT, E., *Optics*, Second Edition, Addison-Wesley, 676 pp. (1987).

P153385.PDF [Page: 171 of 191]

[38] HECHT-NIELSEN, R., *Neurocomputing*, Addison-Wesley, 433 pp. (1990).

[39] HOLLER, M., TAM, S., CASTRO, H., BENSON, R., «An Electrically Trainable Artificial Neural Network (ETANN)», dans *Proc. Intl. Conf. Neural Networks*, pp. II-191-II-196 (1989).

[40] HONG, J., PSALTIS, D. «Storage Capacity of Holographic Associative Memories», dans *Optics Letters*, vol. 12(11), pp. 812-814 (1986).

[41] HOPFIELD, J. J., «Neural Networks and Physical Systems with Emergent Collective Computational Abilities», dans *Proc. Nat. Acad. Sci*, vol. 79, pp. 2554-2558 (1982).

[42] HORIO, Y., NAKAMURA, S., «Analog Memories for VLSI Neurocomputing», dans Artificial Neural Networks-Paradigms, Applications, and Hardware Implementations, éd. Sánchez-Sinencion, E et Lau, C. IEEE Press, 549 pp. (1992).

[43] HORNER, J. L., GIANINO, P. D., «Phase-only matched filtering», dans *Applied Optics*, vol. 23(6), pp. 812-816 (1984).

[44] HSU, K.-H., LI, H.-Y. et PSALTIS, D. «Holographic Implementation of a Fully Connected Neural Network», dans *Proceedings of the IEEE*, vol. 78(10), pp. 1637-1645 (1990).

[45] HUGHES, fiche technique, 2051 Palomar Airport Rd, Carlsbad, CA, 92009 (Février 1989).

[46] ISMAIL, M., FIEZ, T., *Analog VLSI: Signal and Information Processing*, McGraw-Hill, 741 pp. (1994).

P153385.PDF [Page: 172 of 191]

[47] JARED, D. A., JOHNSON, K. M., «Ferroelectric Liquid Crystal Spatial Light Modulators», dans *SPIE Critical Reviews Series*, vol. 1150, pp. 46-60.

[48] JONES, S., SAMMUT, K., NIELSEN, C.H. et STAUNSTRUP, J., «Toroidal Neural Network: Architecture and Processor Granularity Issues», dans *VLSI Design of Neural Networks*, éd Ramacher et Ruckert, pp. 229-254.

[49] KODAK, fiche technique du vidéo projecteur LC500, no. V3-581.

[50] KOSKO, B., «Bidirectional Associative Memories», dans *IEEE Trans. Syst. Man. Cybern.*, vol 18, pp. 49-60, (1988).

[51] KUNG, S. Y., HWANG, J. N., «A Unifying Algorithm/Architecture for Artificial Neural Networks», dans *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, vol. 4, pp. 2505-2508, (1989).

[52] MAHER, M.A.C., DEWEERTH, S.P., MAHOWALD, M.A. et MEAD C.A., «Implementing Neural Architectures Using Analog VLSI Circuits», dans *IEEE Transactions on Circuits and Systems*, vol. 36(5) pp.643-652 (1989).

[53] MANN, J. R., GILBERT, S., «An Analog Self-Organizing Neural Network Chip», dans *Advances in Neural Information Processing Systems 1*, éd. D. Touretzky, pp.739-747, (1989).

[54] MAREN, A., HARSTON, C., PAP, R., *Handbook of Neural Computing Applications*, Academic Press, 448 pp. (1990).

[55] McAULAY, A. D., *Optical Computer Architectures*, Wiley-Interscience, 531 pp. (1991).

P153385.PDF [Page: 173 of 191]

[56] MEAD, C.A., *Analog VLSI and Neural Systems* Addison Wesley, 371 pp. (1989).

[57] MICRO OPTICS, fiche technique, 8608 University Green #5 Middleton, WI 53562 (Juin 1993).

[58] MOK, F. H., TACKITT, M. C., STOLL, H. M., «Storage of 500 high resolution holograms in a $LiNnO_3$ crystal», dans *Optics Letters*, vol. 16(8), pp. 605-607, (1991).

[59] MOSCA, E. P., GRIFFIN, R. D., PURSEL, F. P., LEE, J. N., «Acoustooptical matrix-vector product processor: implementation issues», dans *Applied Optics*, vol. 28(18), pp. 3843-3851 (1989).

[60] MUELLER, P. *et al.* «A Programmable Analog Neural Computer and Simulator», dans *Advances in Neural Information Processing Systems 1*, pp. 712-719 (1989).

[61] MURRAY, A.F., «Pulse Arithmetic in VLSI Neural Networks», dans *IEEE Micro Magazine*, pp. 64-74, (decembre 1989).

[62] OPPENHEIM, A. V., SCHAFER, R. W., *Discrete-Time Signal Processing*, Prentice Hall, 879 pp. (1989).

[63] OWECHKO, Y., «Nonlinear Holographic Associative Memories», dans *IEEE Journal of Quantum Electronics*, vol. 25(3), (1989).

[64] PAEK, E. G., PSALTIS, D., «Optical Associative Memory Using Fourier Transform Holograms», dans *Optical Engineering*, vol. 26(5), pp. 428-433, (1987).

P153385.PDF [Page: 174 of 191]

[65] PENZ, P.A., «The Closeness Code: An Integer to Binary Vector Transformation Suitable For Neural Networks Algorithms», dans *Proc. IEEE Int. Conf. Neural Networks*, vol. 3, pp. 515-22, (1987).

[66] PSALTIS, D., ATHALE, R., «High accuracy computation with linear analog optical systems: a critical study.», *Applied Optics*, vol. 25(18), pp. 3071-3077 (1992).

[67] PSALTIS, D., BRADY, D., WAGNER, K., «Adaptive optical networks using photorefractive crystals», dans *Applied Optics*, vol. 27(9) pp. 1752-1759 (1988).

[68] PSALTIS, D., BRADY, D., GU, X.-G. et HSU, K., «Optical Implementation of Neural Networks», dans Optical Processing and Computing, éd. Arsenault, H. H., *et al.*, pp. 251-276 (1989).

[69] RAMACHER, U., BEICHTER, J., BRULS, N et SICHENEDER, E., «Architecture and VLSI design of a VLSI Neural Signal Processor», dans *IEEE Proc. Int. Symposium on Circuits and Systems*, pp. 1975-1978 (1993).

[70] RAMACHER, U. *et al*, «Design of a 1st Generation Neurocomputer», dans *VLSI Design of Neural Networks*, éd Ramacher et Ruckert, pp. 271-310.

[71] RAND, W.M., «Objective Criteria for the Evaluation of Clustering Methods», dans *Journal Am. Statist. Assoc.*, Vol. 66, pp. 846-850 (1971).

[72] RAO, A., WALKER, M.R., CLARK L.T., et AKERS, L.A., «ART1 Network Implementation Issues», dans *TENCON '89, Fourth IEEE Region 10 International Conference*, pp. 462-6 (1989).

[73] SAVARIA, Y., *Conception et vérification des circuits VLSI*, Éditions de l'École Polytechnique de Montréal, 398 pp. (1988).

P153385.PDF [Page: 175 of 191]

[74] SCHLEHER, D.C., *Introduction to Electronic Warfare*, Artech House, (1986).

[75] SCHWARTZ, D. B., HOWARD, R., HUBBARD, W., «A Programmable Analog Neural Network Chip», dans *IEEE Journal of Solid State Circuits*, vol. 24(3), (1989).

[76] Signal & Image Processing, dans *SPIE's International Technical Working Group Newsletter*, vol. 3(1) p. 3 (mars 1992).

[77] TOMBERG, J.E., KASKI, K.K.K., «Pulse-Density Modulation Technique in VLSI Implementations of Neural Network Algorithms», dans *IEEE Journal of Solid-State Circuits*, vol. 25(5), pp. 1277-1286, (1990).

[78] TSAY, S.W., et NEWCOMB, R.W., «VLSI Implementation of ART1 Memories», in *IEEE Transactions on Neural Networks*, vol. 2, no. 2, pp. 214-221 (1991).

[79] VITTOZ, E. *et al.* «Analog Storage of Adjustable Synaptic Weights», dans *VLSI Design of Neural Networks*, éd Ramacher et Ruckert, pp. 47-63.

[80] WUNSCH, D. C., CAUDELL, T. P., et ZIKAN, K., «Optical and Optoelectronic Neural Network Implementations at the Boeing Company», dans *1992 Government Microcircuit Applications Conference (GOMAC), Digest of Papers*, pp. 559-562.

[81] YASUNAGA, M. *et al.* «Design, Fabrication and Evaluation of a 5-inch Wafer Scale Neural Network LSI Composed of 576 Digital Neurons», dans *Proc. Int. Joint Conf. on Neural Networks*, vol. II, pp.527-535 (1990).

ANNEXE A PROPOSITION DE PROJET D'UN RÉSEAU DE NEURONES ART1 INTÉGRÉ

Ce rapport propose une nouvelle architecture de réseau de neurones intégré. Le réseau concerné est une version modifiée du populaire réseau Adaptive Resonance Theory (ART). La modification touche une des deux lois d'apprentissage inhérente au réseau. Cette modification permet une implantation ITGE plus efficace. Ces modifications sont décrites en détail dans le chapitre V et dans la référence [16]. Une équipe de travail, constituée de quatre étudiants gradués, a développé une puce suivant les concepts proposés dans le présent rapport.

A.1 Introduction

Le réseau ART (Adaptive Resonance Theory) a été inventé par les docteurs Stephen Grossberg et Gail A. Carpenter de l'Université de Boston. Initialement, les deux chercheurs s'étaient questionnés sur le problème qu'ils ont par la suite baptisé "le problème de la stabilité et de la plasticité". Comment notre cerveau est-il capable de reconnaître un stimulus, même après plusieurs années? Par exemple, dans notre jeunesse, on nous a appris ce qu'était une chaise, et même à soixante ans, nous saurons toujours ce qu'est une chaise. Cette propriété de mémoire à long terme porte le nom de "stabilité". Comment se fait-il que, lorsqu'on nous présente un fauteuil, nous sommes en mesure de dire qu'il s'agit en effet d'un fauteuil et non d'une chaise; les deux se ressemblent bien pourtant? Cette propriété

P153385.PDF [Page: 177 of 191]

qu'a le cerveau de ne pas corrompre la mémoire existante pour accommoder de nouveaux stimuli porte le nom de "plasticité".

La stabilité et la plasticité sont donc deux propriétés essentielles dans tout réseau de neurones. Pourtant, le plus populaire des réseaux à apprentissage supervisé, le *backpropagation*, souffre du problème de plasticité. En effet, après le long apprentissage caractéristique de ce type de réseaux, si un nouveau stimulus est présenté au réseau lorsqu'il est toujours en mode d'apprentissage, la mémoire existante sera corrompue à un point tel que les stimuli déjà appris seront oubliés. C'est un peu comme si un enfant allait passer quelques semaines dans un camp de vacances et à son retour, serait incapable de reconnaître ses parents.

Il existe plusieurs types de réseaux ART: ART1, ART2, ART3, ARTMAP, FUZZY ART et plus récemment FUZZY ARTMAP. Le type de réseau qui nous intéresse ici est ART1. Ce dernier est utilisé principalement pour la reconnaissance et la classification de signaux binaires.

A.2 Principe de fonctionnement

Dans cette section, nous allons décrire le fonctionnement d'un réseau ART1 modifié [16]. Le lecteur qui désire connaître le fonctionnement du réseau ART1 initial, tel que décrit par Carpenter et Grossberg, est invité à consulter la référence [10].

Le réseau ART1 est un réseau de classification sans supervision constitué d'une couche de neurones d'entrée et une couche de neurones de sortie, toutes deux identifiées respectivement par F1 et F2 (voir figure A.1). Le terme "sans supervision" signifie que le réseau ne nécessite aucune aide du monde extérieur pour effectuer sa classification: il n'utilise que l'information contenue dans les données pour former les différentes classes.

P153385.PDF [Page: 178 of 191]

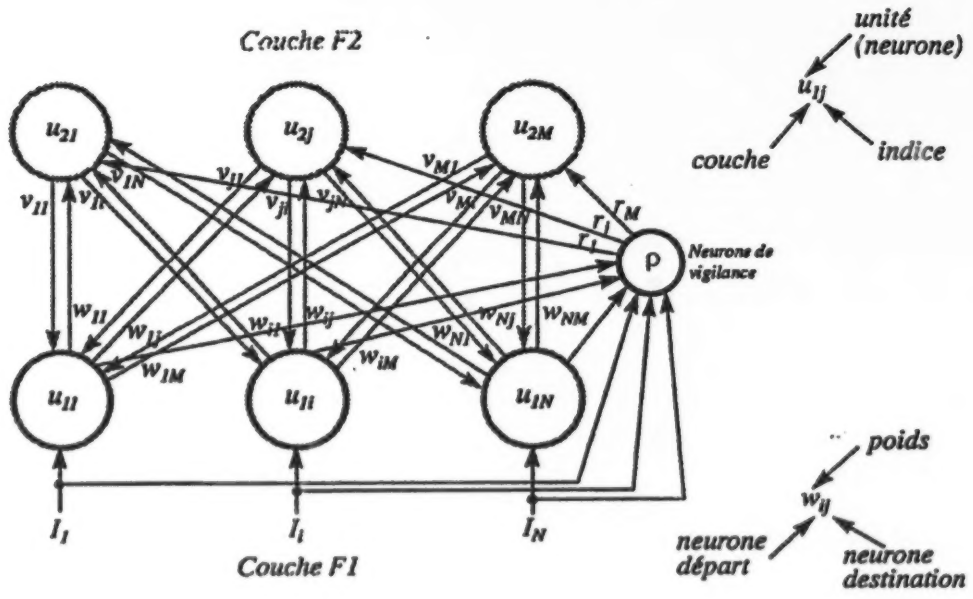


Figure A.1: Schéma du réseau ART1.

Les patrons d'entrée sont présentés aux neurones de la couche F1. Ces derniers propagent le patron sur tous les neurones de la couche F2. Suite à la présentation d'un patron **I**, tous les neurones de la couche F2 calculent leur *entrant net*, défini par l'équation suivante:

$$T_j = \sum_{i=1}^{N} I_i w_{ij} + \Theta_j, \tag{A.1}$$

où w_{ij} dénote le poids bas-vers-haut (BH) reliant le neurone F1 i au neurone F2 j. Nous reviendrons plus tard sur la signification du biais Θ_j. Les neurones de la couche F1 entrent alors en compétition pour déterminer lequel d'entre eux possède l'entrant net le plus élevé. Le neurone gagnant, d'indice J, envoie alors son prototype $\mathbf{V}_J$ à la couche F1 pour que le calcul du *test de vigilance* prenne place. Le prototype $\mathbf{V}_J$ correspond aux N poids haut-bas (HB) v_{Ji}, $i = 1, 2, \ldots, N$, émanant du neurone F2 d'indice J vers les N neurones de la couche F1.

P153385.PDF [Page: 179 of 191]

Le but du test de vigilance est de calculer la similarité entre le prototype $\mathbf{V}_J$ et le patron d'entrée $\mathbf{I}$ pour ensuite la confronter à un seuil ρ appelé *seuil de vigilance* ou *critère de vigilance* selon l'équation

$$\frac{\|\mathbf{I} \cap \mathbf{V}_J\|}{\|\mathbf{I}\|} \geq \rho, \tag{A.2}$$

où $\mathbf{I} \cap \mathbf{V}_J$ est l'intersection (ET-binaire) entre $\mathbf{I}$ et $\mathbf{V}_J$ et $\|\cdot\|$ est l'opérateur "norme", c'est-à-dire $\|\mathbf{X}\| = \sum_{i=1}^{N} x_i$.

Si le prototype réussit à passer le test de vigilance, on dira que le neurone J entre en *résonance*, période durant laquelle il ajuste ses poids $\mathbf{V}_J$ et $\mathbf{W}^{(J)}$ pour *apprendre* le patron $\mathbf{I}$ ($\mathbf{W}^{(J)} = \{w_{iJ} | i = 1, 2, \ldots N\}$). Sinon, le réseau désactivera temporairement le neurone J et engagera une nouvelle compétition pour déterminer un nouveau prototype candidat. Ce processus de recherche itératif se poursuivra jusqu'à ce qu'il y ait résonance ou jusqu'à ce qu'une nouvelle classe soit créée en assignant le patron $\mathbf{I}$ à un neurone F2 non-commis. Le réseau est alors prêt à classifier un nouveau patron d'entrée.

La désactivation des neurones F2 est contrôlée par le *neurone de vigilance* (voir figure A.1). Ce dernier émet un signal r_j à chaque neurone F2. Si $r_j = 1$, le neurone j sera désactivé jusqu'à la présentation d'un nouveau patron d'entrée. À la fin du cycle de reconnaissance, tous les signaux r_j sont initialisés à zéro.

La mise à jour des poids $\mathbf{V}_J$ et $\mathbf{W}^{(J)}$, suite à une résonance, est régie d'une part par la *loi d'apprentissage des prototypes* (template learning rule, c.f. [10]) définie par

$$\mathbf{V}_J' = \mathbf{I} \cap \mathbf{V}_J, \tag{A.3}$$

et d'autre part par la *loi de codage direct* (direct coding rule, c.f. [16]), définie par

$$\mathbf{W}^{(J)} = (\mathbf{I} \cap \mathbf{V}_J)\,(\Psi - \|\mathbf{I} \cap \mathbf{V}_J\|) = \mathbf{V}_J'\,(\Psi - \|\mathbf{V}_J'\|), \tag{A.4}$$

P153385.PDF [Page: 180 of 191]

où le "prime" dénote la version mise-à-jour du vecteur de poids et Ψ est une constante telle que $\Psi > 2N - 1$ [16].

L'entrant net de l'équation (A.1) peut être simplifié avec l'équation (A.3) et (A.4) comme suit:

$$T_j = \sum_{i=1}^{N} I_i w_{ij} + \Theta_j \tag{A.5}$$

$$= \{ \sum_{i=1}^{N} I_i v_{ji} (\Psi - \|\mathbf{V}_j\|) \} + \Theta_j \tag{A.6}$$

$$= (\Psi - \|\mathbf{V}_j\|) \sum_{i=1}^{N} I_i v_{ji} + \Theta_j \tag{A.7}$$

$$T_j = \|\mathbf{I} \cap \mathbf{V}_j\| (\Psi - \|\mathbf{V}_j\|) + \Theta_j \tag{A.8}$$

où nous avons utilisé le fait que $\sum_{i=1}^{N} I_i v_{ji} = \|\mathbf{I} \cap \mathbf{V}_j\|$. La valeur du biais dépend du patron d'entrée $\mathbf{I}$ et du prototype $\mathbf{V}_j$:

$$\Theta_j = \begin{cases} \alpha > N^2 - 2N + 1 & \text{si } \mathbf{V}_j \text{ est un sous-ensemble de } \mathbf{I} (\mathbf{V}_j \subseteq \mathbf{I}), \\ 0 & \text{autrement.} \end{cases} \tag{A.9}$$

Le prototype $\mathbf{V}_j$ sera un sous-ensemble de $\mathbf{I}$ si $\forall v_{ji} = 1, I_i = 1; i = 1, \ldots, N$. Par exemple, à la figure A.2, le prototype est un sous-ensemble du patron d'entrée.

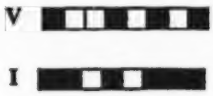


Figure A.2: Exemple de sous-ensemble.

P153385.PDF [Page: 181 of 191]

Dans ce qui suit, nous allons assumer, sans perte de généralité, que

$$\Theta_j = \begin{cases} N^2 & \mathbf{V}_j \subseteq \mathbf{I}, \\ 0 & \text{autrement.} \end{cases} \tag{A.10}$$

L'algorithme que nous venons de décrire est résumé sous la forme d'un organigramme à la figure A.3 de la page suivante.

A.3 Architecture proposée

La description du principe de fonctionnement du réseau nous permet de décomposer l'effort de calcul en quatre principales tâches:

- Calcul de l'entrant net;
- Détermination du gagnant;
- Calcul du test de vigilance;
- Mise à jour des poids.

Le projet proposé porte sur l'implantation d'un processeur neuronal spécialisé accomplissant la première des quatre tâches, soit le calcul de l'entrant net (équation A.8)). Le reste de cette section décrira une architecture possible pour ce processeur.

A.3.1 Architecture et fonctionnement du circuit

Dans cette section, nous allons décrire brièvement le fonctionnement du circuit proposé. À un niveau hiérarchique supérieur, ce circuit peut être représenté par une boîte noire, telle qu'illustrée à la figure A.4 suivante.

P153385.PDF [Page: 182 of 191]

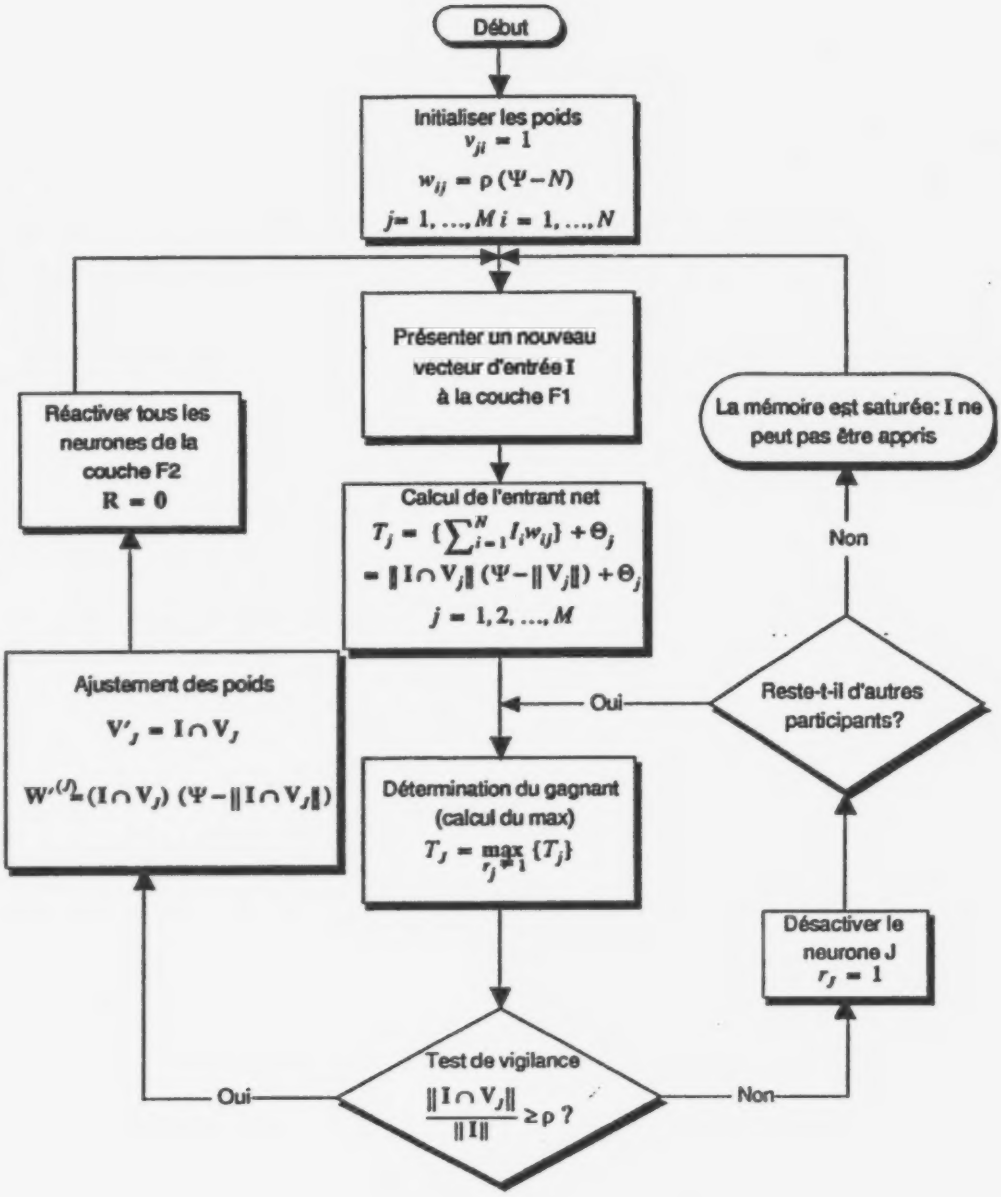


Figure A.3: Organigramme de l'algorithme ART1 modifié

P153385.PDF [Page: 183 of 191]

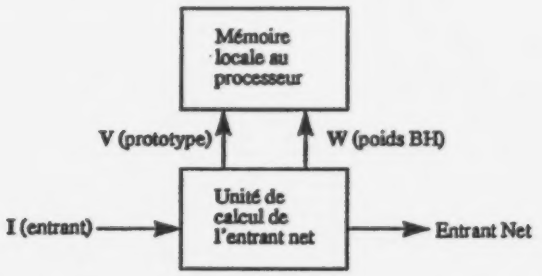


Figure A.4: Diagramme fonctionnel du processeur.

Comme la dimension du vecteur d'entrée **I** peut devenir considérablement grande dans certaines applications (par exemple, 120 bits (voir [16])), il est essentiel de subdiviser ce dernier en plusieurs mots de longueur raisonnable (8 ou 16 bits, par exemple). Ainsi, les données **I** et **V** sont transmises à l'unité de calcul mot par mot. Les mots du patron d'entrée proviendront de l'extérieur de la puce alors que ceux du prototype proviendront d'une mémoire interne locale au processeur. Les dimensions de cette mémoire, à savoir la profondeur et la largeur des mots, restent à déterminer.

La figure A.5 décrit le circuit avec plus de détails. Le gros du travail du processeur est de calculer la norme (i.e. compter le nombre de "1") d'une combinaison entre **I** et **V**. Cette combinaison est en fait l'intersection ("ET" logique) entre les deux vecteurs. Le rôle de la porte "ET" multiple est donc de calculer cette intersection. Le résultat est alors transmis à l'encodeur dont la tâche est de coder le nombre de "1" dans le mot à son entrée dans le système binaire classique. Par exemple, le mot 10110111 serait codé en 0110 puisqu'il comporte 6 "1". Nous anticipons que l'encodeur soit le goulot d'étranglement du processeur. Par conséquent, une attention particulière devra être portée au design de ce module. Si n est le nombre de bits à l'entrée de l'encodeur, la sortie devra comporter $\log_2 n + 1$. Le "+1" est nécessaire pour encoder le vecteur constitué exclusivement de "1".

P153385.PDF [Page: 184 of 191]

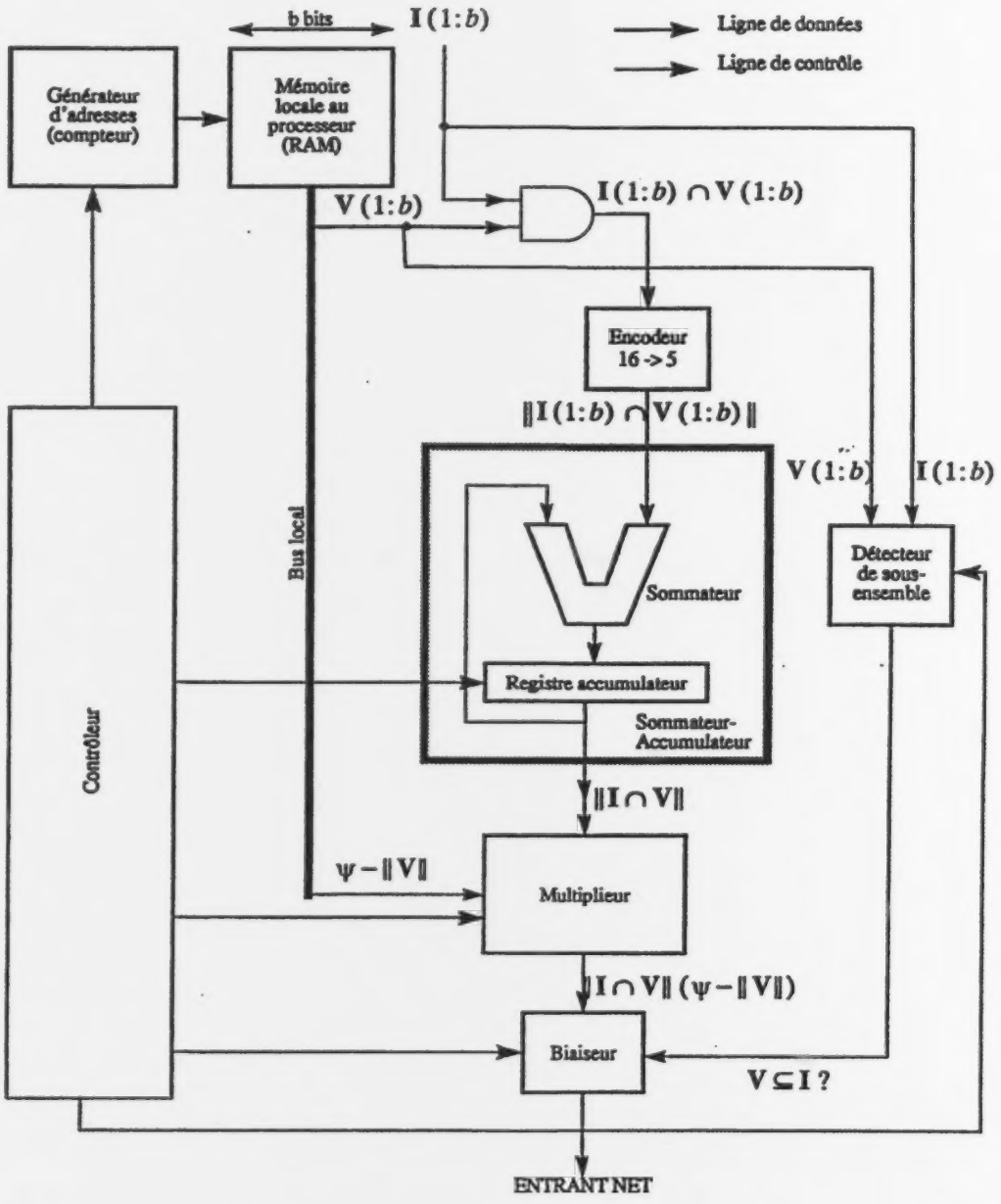


Figure A.5: Architecture du processeur neuronal.

P153385.PDF [Page: 185 of 191]

Le mot binaire encodé est ensuite présenté au sommateur-accumulateur dont le rôle est de calculer, après plusieurs cycles, le nombre total de "1". Cette tâche est accomplie en additionnant successivement les sommes partielles en provenance de l'encodeur. Lorsque tous les mots ont été présentés, le résultat de $\|\mathbf{I} \cap \mathbf{V}\|$ est présent à la sortie de l'encodeur.

La tâche suivante est de multiplier $\|\mathbf{I} \cap \mathbf{V}\|$ avec $(\Psi - \|\mathbf{V}\|)$. La valeur de $(\Psi - \|\mathbf{V}\|)$ est lue de la mémoire locale (elle se trouve immédiatement après le dernier mot de V). Elle est alors transmise par le bus local du processeur jusqu'au multiplieur. Ce dernier effectue le produit $\|\mathbf{I} \cap \mathbf{V}\| (\Psi - \|\mathbf{V}\|)$. Le choix du design du multiplieur, à savoir s'il doit être parallèle ou séquentiel, est un sujet ouvert. Il faudra attendre d'avoir une première estimée de l'espace en silicium occupé par le reste du processeur avant de prendre une décision. De plus, si la rapidité n'est pas justifiable (i.e. si le multiplieur n'est pas le goulot d'étranglement), alors un design séquentiel sera plus approprié.

L'unité dénotée "biaiseur" a pour utilité d'introduire un biais au résultat de la multiplication si le patron d'entrée V est un sous-ensemble de I (i.e. $\mathbf{V} \subseteq \mathbf{I}$). Donc, le module appelé "Détecteur de sous-ensemble" a pour fonction d'envoyer au biaiseur un signal informant si $\mathbf{V} \subseteq \mathbf{I}$.

Le design du biaiseur sera extrêmement simple car la théorie (voir [16]) permet que ce biais soit constant et soit également une puissance de 2. De plus, cette puissance de deux est supérieure à la plus grande puissance de deux du nombre à biaiser (i.e. $\|\mathbf{I} \cap \mathbf{V}\| (\Psi - \|\mathbf{V}\|)$). Le biaiseur devra donc tout simplement basculer le bit immédiatement à gauche du bit le plus significatif du nombre si $\mathbf{V} \subseteq \mathbf{I}$. Il n'est donc pas nécessaire d'utiliser un additionneur. La figure A.6 illustre un exemple.

P153385.PDF [Page: 186 of 191]

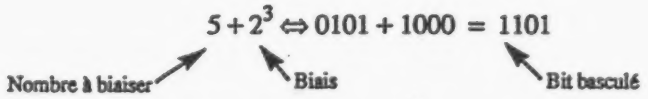


Figure A.6: Introduction du biais.

A.3.2 Contrôle du processeur

Un module de contrôle sera nécessaire pour synchroniser les différents modules avec le flot des données dans le pipeline. Ce module permettra également de procurer au système une certaine flexibilité, notamment dans la longueur des vecteurs d'entrée permissible. Par exemple, si le vecteur d'entrée fait 256 bits et que la largeur des mots de la mémoire est de $b = 16$ bits, le contrôleur devra lire la mémoire et charger le pipeline $256/16 = 16$ fois.

Cet exemple nous montre qu'avec l'information du nombre de bits N et de la largeur de la mémoire b, le contrôleur sera en mesure de déterminer le nombre de cycles requis par chaque module ainsi que la largeur des mots qui circulent dans le datapath. En se basant sur le schéma de la figure A.5, il est possible de dresser le tableau A.1.

Tableau A.1:Nombre de cycles requis et largeur des mots pour les six modules de l'architecture.

Module	Nombre de cycles	Largeur des mots
Mémoire	• N/b cycles de lecture pour lire tous les mots du vecteur • 1 cycle pour lire le poids $\Psi - \|\|V\|\|$ (si la largeur des mots $b \geq \log_2 N + 2$)	b bits
Encodeur	aucun: module combinatoire	• b bits à l'entrée • $\log_2 b + 1$ à la sortie
Sommateur-accumulateur	N/b cycles pour produire $\|\|I \cap V\|\|$	• $\log_2 b + 1$ bits à l'entrée • $\log_2 N + 1$ bits à la sortie
Détecteur de sous-ensemble	N/b cycles	• b bits à l'entrée • 1 bit à la sortie
Multiplieur	$\log_2 N + 1$ cycles	• $\log_2 N + 1$ bits pour le multiplicateur • $\log_2 N + 2$ bits pour le multiplicande • $2(\log_2 N + 1)$ bits pour le produit
Biaiseur	aucun: module combinatoire	• $2(\log_2 N + 1)$ bits à l'entrée • $2(\log_2 N + 1)$ bits à la sortie

P153385.PDF [Page: 187 of 191]

Sachant que le détecteur de sous-ensemble fonctionne en parallèle avec le sommateur-accumulateur, le nombre total de cycles requis avant d'avoir un résultat, la *latence*, est donnée par

$$l = \frac{N}{b} + (\log_2 N + 1)\,. \tag{A.11}$$

Quant au débit, il est déterminé par le module le plus lent dans le pipeline. Pour une valeur de N suffisamment grande, le module le plus lent sera le sommateur-accumulateur. Donc, en première approximation, le débit est donné par

$$D = \frac{b}{N}\,. \tag{A.12}$$

Donc, pour augmenter le débit, il faut soit travailler avec des mots larges, soit avec des petits vecteurs d'entrée, soit une combinaison des deux. Une partie du travail sera de déterminer les valeurs de N et b qui résulteront en un système équilibré.

A.3.3 Modules à réaliser

Les membres de l'équipe auront donc pour mandat de concevoir les modules constituants de l'architecture de la figure A.5. Ces modules sont regroupés dans le tableau suivant.

Tableau A.2:Modules à réaliser.

Module	Fonction	Effort de travail suggéré
Encodeur binaire	Calculer le nombre de "1" dans un mot d'un vecteur binaire ($\|\| \mathbf{I}(1:b) \cap \mathbf{V}(1:b) \|\|$).	25%
Sommateur-accumulateur	Calculer le nombre total de "1" dans un vecteur binaire ($\|\| \mathbf{I} \cap \mathbf{V} \|\|$).	15%
Multiplieur	Calculer le produit $\|\| \mathbf{I} \cap \mathbf{V} \|\| \, (\Psi - \|\| \mathbf{V} \|\|)$.	20%
Détecteur de sous-ensemble	Détecter si $\mathbf{V} \subseteq \mathbf{I}$ et transmettre cette information au biaiseur.	10%
Biaiseur	Ajout du biais Θ si $\mathbf{V} \subseteq \mathbf{I}$.	5%
Contrôleur	Contrôle la synchronisation des modules	25%

P153385.PDF [Page: 188 of 191]

A.3.4 Méthode de conception

La technologie proposée pour la réalisation du processeur est la CMOS4S @ 1,2 µm. Le processeur sera entièrement numérique (aucune composante analogique). Nous anticipons que la majeure partie du design se fera au niveau "porte" et que les modules seront compilés par placement-routage automatique. Cependant, il se peut que certains modules critiques comme l'encodeur nécessitent un développement au niveau masque pour minimiser le délai.

La puce à concevoir comportera un seul processeur. La puce finale, quant à elle, comportera plusieurs processeurs fonctionnant tous en parallèle.

Le processeur devra, bien entendu, être testable. Nous laissons le choix de la stratégie de test à l'équipe de travail. Il serait aussi important de produire une cellule normalisée à partir du processeur pour faciliter l'intégration du reste du système par une équipe ultérieure.

A.4 Conclusion

Ce rapport a proposé une nouvelle architecture pour le réseau de neurones ART1. Cette architecture a été développée dans le cadre du cours ELE6305. Le projet consiste au développement d'un processeur neuronal spécialisé dans le calcul de l'entrant net d'un neurone, opération très coûteuse pour les réseaux ART. L'architecture du processeur est de type pipeline, dans le but de maximiser le débit.

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM
(highest classification of Title, Abstract, Keywords)

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

ECOLE POLYTECHNIQUE DE MONTREAL
P.O. BOX 6079, STATION CENTRE-VILLE
MONTREAL, QUEBEC H3C 3A7 CANADA

2. SECURITY CLASSIFICATION (overall security classification of the document. including special warning terms if applicable)

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

NEURAL NETWORK IMPLEMENTATION STRATEGIES: A COMPARATIVE STUDY (U)

4. AUTHORS (Last name, first name, middle initial)

CRESPO, JEAN-FRANCOIS AND SAVARIA, YVON

5. DATE OF PUBLICATION (month and year of publication of document)

NOVEMBER 1994

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)

182

6b. NO. OF REFS (total cited in document)

81

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

FINAL REPORT

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
NATIONAL DEFENCE
SHIRLEYS BAY, OTTAWA, ONTARIO K1A 0Z4 CANADA

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)

R&D PROJECT 011LB11

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

W7714-2-9630/01-ST

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

P1708

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

95-602

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(XX) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availabilty (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

UNLIMITED

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM

DCD03 2/06/87

P153385.PDF [Page: 190 of 191]

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both offical languages unless the text is bilingual).

(U) This work is divided into two parts. Part one is a comparative study of the implementation technologies for artificial neural networks. The technologies considered are optics and electronics. For the optics technology, we consider optical algebraic processors based architectures, correlator based architectures and holographic interconnection based architectures. For the electronics technology, we consider digital and analog architectures. Both technologies are confronted on the basis of performance, technological and practicality criterias. Optics allows for the highest performances for correlator based architectures which take full advantage of the superior interconnecting power of that technology. However, the limited maturity of optics hinders commercialization. Digital electronics outperforms the other architectures ecause of its flexibility and the accuracy of computation it provides. It is also the most successfully commercialized approach. However, subthreshold analog electronics is the most power-efficient approach.

(U) In part two, a new learning law, the Direct Coding Rule, is proposed for bottom-up long term memory learning in Adaptive Resonance Theory (ART) networks. This law requires less computational procesion than the traditional Weber Law Rule and modifies the search dynamics of the network to accelerate convergence. Following a brief mathematical analysis of the new learning law, an ART1 network based on this law is applied to a passive radar detection problem. Simulation results show that for a quantization level of 7 bits in the weights, the Direct Coding Rule speeds up convergence by a factor of 10 compared to the Weber Law Rule.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

FRENCH LANGUAGE
NEURAL NETWORKS
IMPLEMENTATION STRATEGIES
VLSI ARCHITECTURES
OPTICS
ADAPTIVE RESONANCE THEORY (ART)